

# **Penugasan Latex Aplikom**

**Latek Aplikom/Tugas**

oleh

**Nasywa Fierda Azizah**

**23030630068**



**PROGRAM STUDI SARJANA MATEMATIKA  
DEPARTEMEN PENDIDIKAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS NEGERI YOGYAKARTA**

**2023**



# **PENUGASAN LATEX APLIKOM**

## **Tugas Latex Aplikom/Tugas**

Sebagai syarat memenuhi tugas aplikom  
Aplikasi Komputer

oleh  
**Nasywa Fierda Azizah**  
**23030630068**

**PROGRAM STUDI SARJANA MATEMATIKA  
DEPARTEMEN PENDIDIKAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS NEGERI YOGYAKARTA**

**2023**





## PENDAHULUAN

Saya yang membuat ini:

Nama : Nasywa Fierda Azizah  
NIM : 23030630068  
Departemen : Matematika  
Penulis tugas berjudul : Tugas Latex Aplikom  
dengan ini menyatakan bahwa:

1. Tugas ini untuk memenuhi tugas aplikom, selain itu untuk menambah wawasan,
2. Tujuannya di kemudian hari tugas yang saya tulis dapat berguna, untuk dipelajari kembali.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Yogyakarta, 29 November 2024  
Yang menyatakan,

Nasywa Fierda Azizah  
23030630068



## DAFTAR ISI

HALAMAN JUDUL. . . . .	i
PENDAHULUAN . . . . .	iii
DAFTAR ISI . . . . .	v
DAFTAR TABEL . . . . .	vii
DAFTAR GAMBAR. . . . .	ix
DAFTAR LAMPIRAN . . . . .	xi
 BAB I     Statistika. . . . .	 1
EMT4Statistika_Nasywa Fierda Azizah_23030630068. . . . .	1
EMT untuk Statistika . . . . .	3
Tabel . . . . .	7
Distribusi. . . . .	13
Distribusi Diskrit . . . . .	17
Merencanakan Data . . . . .	21
Regresi dan Korelasi. . . . .	29
Membuat Fungsi baru. . . . .	33
Simulasi Monte Carlo . . . . .	35
Tes . . . . .	39
Beberapa Tes Lagi . . . . .	43
Angka Acak . . . . .	47
Pengantar untuk Pengguna Proyek R . . . . .	49
Sintaks Dasar . . . . .	51
Pengindeksan . . . . .	55
Tipe Data. . . . .	57
Faktor dan Tabel. . . . .	59
Array . . . . .	63
Daftar . . . . .	67
File Input dan Output (Membaca dan Menulis Data) . . . . .	69
File CSV . . . . .	73
Menggunakan Tabel. . . . .	77
Menganalisis Garis . . . . .	79
Membaca dari Web . . . . .	81
Input dan Output Variabel . . . . .	83
Plot 2D . . . . .	85

<b>nasywa plot 2. . . . .</b>	<b>1</b>
<b>Menggambar Grafik 2D dengan EMT . . . . .</b>	<b>3</b>
Basic Plots . . . . .	3
Plot Aspect . . . . .	5
<b>Plot 2D di Euler . . . . .</b>	<b>7</b>
<b>Plot Ekspresi atau Variabel . . . . .</b>	<b>9</b>
<b>Fungsi dalam satu Parameter. . . . .</b>	<b>15</b>
<b>Menggambar Beberapa Kurva pada bidang koordinat yang</b>	
<b>sama . . . . .</b>	<b>19</b>
Soal Latihan Tambahan . . . . .	29
Label Teks . . . . .	30
LaTeX . . . . .	33
Interaksi Pengguna . . . . .	35
Gaya Plot 2D . . . . .	38
<b>Memplot Data 2D . . . . .</b>	<b>47</b>
<b>Menggambar Daerah Yang Dibatasi Kurva . . . . .</b>	<b>51</b>
<b>Grafik Fungsi Parametrik. . . . .</b>	<b>57</b>
<b>Menggambar Grafik Bilangan Kompleks . . . . .</b>	<b>59</b>



## DAFTAR TABEL



## **DAFTAR GAMBAR**



## **DAFTAR LAMPIRAN**



## BAB I

### STATISTIKA

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam

vulputate metus eu enim. Vestibulum pellentesque felis eu massa.  
unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex  
[T1]fontenc [utf8]inputenc textcomp lmodern upquote [microtype  
[protrusion]basicmath parskip bookmark xurl



**EMT4STATISTIKA\_NASYWA FIERDA AZIZAH\_23030630068**

Nama : Nasywa Fierda Azizah

NIM : 23030630068

Kelas : Matematika B



## EMT UNTUK STATISTIKA

Di buku catatan ini, kami mendemonstrasikan plot statistik utama, pengujian, dan distribusi di Euler.

Mari kita mulai dengan beberapa statistik deskriptif. Ini bukan pengantar statistik. Jadi, Anda mungkin memerlukan latar belakang untuk memahami detailnya.

Asumsikan pengukuran berikut. Kami ingin menghitung nilai rata-rata dan deviasi standar yang diukur.

```
>M=[1000,1004,998,997,1002,1001,998,1004,998,997]; ...  
> median(M), mean(M), dev(M),
```

```
Function median not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... 04,998,997,1002,1001,998,1004,998,997]; median(M)
```

Kita dapat memplot plot kotak-dan-kumis untuk datanya. Dalam kasus kami, tidak ada outlier.

```
>aspect(1.75); boxplot(M):
```

```
Function boxplot not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
aspect(1.75); boxplot(M) : ...
```

Kami menghitung probabilitas suatu nilai lebih besar dari 1005, dengan asumsi nilai terukur berdistribusi normal.

Semua fungsi untuk distribusi di Euler diakhiri dengan ...dis dan menghitung distribusi probabilitas kumulatif (CPF).

Kami mencetak hasilnya dalam % dengan akurasi 2 digit menggunakan fungsi print.

```
>print((1-normaldis(1005,mean(M),dev(M)))*100,2,unit=""  
%")
```

```
Function mean not found.
```

```
Try list ... to find functions!
Error in:
print((1-normaldis(1005,mean(M),dev(M)))*100,2,unit=" %")
^
```

Untuk contoh berikutnya, kita asumsikan jumlah pria berikut dalam rentang ukuran tertentu.

```
>r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Berikut adalah alur pendistribusiannya.

```
>plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="\/"):
^
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... lot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="\/"): ...
^
```

Kita bisa memasukkan data mentah tersebut ke dalam tabel.

Tabel adalah metode untuk menyimpan data statistik. Tabel kita harus berisi tiga kolom: Awal jangkauan, akhir jangkauan, jumlah pria dalam jangkauan.

Tabel dapat dicetak dengan header. Kami menggunakan vektor string untuk mengatur header.

```
>T:=r[1:8]' | r[2:9]' | v'; writetable(T,labc=["BB","BA","Frek"])
```

```
Function writetable not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... 2:9]' | v'; writetable(T,labc=["BB","BA","Frek"]) ...
^
```

Jika kita memerlukan nilai rata-rata dan statistik ukuran lainnya, kita perlu menghitung titik tengah rentang tersebut. Kita bisa menggunakan dua kolom pertama tabel kita untuk ini.

Symbol “|” digunakan untuk memisahkan kolom, fungsi “writetable” digunakan untuk menulis tabel, dengan opsi “labc” untuk menentukan header kolom.

```
>(T[,1]+T[,2])/2 // the midpoint of each interval
```

```
157.5
```

```
161.5
```

```

165.5
169.5
173.5
177.5
181.5
185.5

```

Namun akan lebih mudah jika menjumlahkan rentang dengan vektor  $[1/2, 1/2]$ .

```

>M=fold(r,[0.5,0.5])
[157.5, 161.5, 165.5, 169.5, 173.5, 177.5, 181.5]

```

Sekarang kita dapat menghitung mean dan deviasi sampel dengan frekuensi tertentu.

```

>{m,d}=meandev(M,v); m, d,
Function meandev not found.
Try list ... to find functions!
Error in:
{m,d}=meandev(M,v); m, d, ...
^

```

Mari kita tambahkan distribusi nilai normal ke diagram batang di atas. Rumus distribusi normal dengan mean  $m$  dan simpangan baku  $d$  adalah:

Karena nilainya antara 0 dan 1, maka untuk memplotnya pada bar plot harus dikalikan dengan 4 kali jumlah data.

```

>plot2d("qnormal(x,m,d)*sum(v)*4",...
> xmin=min(r),xmax=max(r),thickness=3,add=1):
Function plot2d not found.
Try list ... to find functions!
Error in:
... *4",    xmin=min(r),xmax=max(r),thickness=3,add=1)

```



## TABEL

Di direktori buku catatan ini Anda menemukan file dengan tabel. Data tersebut merupakan hasil survei. Berikut adalah empat baris pertama file tersebut. Datanya berasal dari buku online Jerman “Einführung in die Statistik mit R” oleh A. Handl.

```
>printfile("table.dat",4);  
Function printfile not found.  
Try list ... to find functions!  
Error in:  
printfile("table.dat",4); ...  
      ^
```

Tabel berisi 7 kolom angka atau token (string). Kami ingin membaca tabel dari file. Pertama, kami menggunakan terjemahan kami sendiri untuk tokennya.

Untuk ini, kami mendefinisikan kumpulan token. Fungsi `strtokens()` mendapatkan vektor string token dari string tertentu.

```
>mf=["m","f"]; yn=["y","n"]; ev:=strtokens("g vg m b vb");  
Sekarang kita membaca tabel dengan terjemahan ini.
```

Argumen `tok2`, `tok4` dll. adalah terjemahan dari kolom tabel. Argumen ini tidak ada dalam daftar parameter `readtable()`, jadi Anda perlu menyediakannya dengan “:=”.

```
>{MT,hd}=readtable("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);  
Function readtable not found.  
Try list ... to find functions!  
Error in:  
... ("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn)
```

```
>load over statistics;  
Could not open statistics!  
Error in:  
load over statistics; ...  
      ^
```

Untuk mencetak, kita perlu menentukan kumpulan token yang sama. Kami mencetak empat baris pertama saja.

```
>writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
MT is not a variable!
Error in:
writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=e
^
```

Titik “.” mewakili nilai-nilai, yang tidak tersedia.

Jika kita tidak ingin menentukan token yang akan diterjemahkan terlebih dahulu, kita hanya perlu menentukan, kolom mana yang berisi token dan bukan angka.

```
>ctok=[2,4,5,7]; {MT,hd,tok}=readtable("table.dat",ctok=ctok);
Function readtable not found.
Try list ... to find functions!
Error in:
... ,7]; {MT,hd,tok}=readtable("table.dat",ctok=ctok); ...
^
```

Fungsi readtable() kini mengembalikan sekumpulan token.

```
>tok
Variable tok not found!
Error in:
tok ...
^
```

Tabel berisi entri dari file dengan token yang diterjemahkan ke dalam angka.

String khusus NA = “.” diartikan sebagai “Tidak Tersedia”, dan mendapatkan NAN (bukan angka) di tabel. Terjemahan ini dapat diubah dengan parameter NA, dan NAval.

```
>MT[1]
MT is not a variable!
Error in:
MT[1] ...
^
```

Berikut isi tabel dengan nomor yang belum diterjemahkan.

```
>writetable(MT,wc=5)
```



```
Variable or function MT not found.
Error in:
writetable(MT,wc=5) ...
      ^
```

Untuk kenyamanan, Anda dapat memasukkan keluaran `readtable()` ke dalam daftar.

```
>Table={{readtable("table.dat",ctok=ctok)}};
Function readtable not found.
Try list ... to find functions!
Error in:
Table={{readtable("table.dat",ctok=ctok)}}; ...
      ^
```

Dengan menggunakan kolom token yang sama dan token yang dibaca dari file, kita dapat mencetak tabel. Kita dapat menentukan `ctok`, `tok`, dll. atau menggunakan tabel daftar.

```
>writetable(Table,ctok=ctok,wc=5);
Variable or function Table not found.
Error in:
writetable(Table,ctok=ctok,wc=5); ...
      ^
```

Fungsi `tablecol()` mengembalikan nilai kolom tabel, melewati baris apa pun dengan nilai NAN ("." dalam file), dan indeks kolom, yang berisi nilai-nilai ini.

```
>{c,i}=tablecol(MT,[5,6]);
Variable or function MT not found.
Error in:
{c,i}=tablecol(MT,[5,6]); ...
      ^
```

Kita bisa menggunakan ini untuk mengekstrak kolom dari tabel untuk tabel baru.

```
>j=[1,5,6]; writetable(MT[i,j],labc=hd[j],ctok=[2],tok=tok)
Variable or function i not found.
Error in:
j=[1,5,6]; writetable(MT[i,j],labc=hd[j],ctok=[2],tok=tok)
      ^
```

Tentu saja, kita perlu mengekstrak tabel itu sendiri dari daftar Tabel dalam kasus ini.

```
>MT=Table[1];  
Table is not a variable!  
Error in:  
MT=Table[1]; ...  
^
```

Tentu saja, kita juga dapat menggunakannya untuk menentukan nilai rata-rata suatu kolom atau nilai statistik lainnya.

```
>mean(tablecol(MT,6))  
Variable or function MT not found.  
Error in:  
mean(tablecol(MT,6)) ...  
^
```

Fungsi `getstatistics()` mengembalikan elemen dalam vektor, dan jumlahnya. Kami menerapkannya pada nilai “m” dan “f” di kolom kedua tabel kami.

```
>{xu,count}=getstatistics(tablecol(MT,2)); xu, count,  
Variable or function MT not found.  
Error in:  
{xu,count}=getstatistics(tablecol(MT,2)); xu, count, ...  
^
```

Kita bisa mencetak hasilnya di tabel baru.

```
>writetable(count',labr=tok[xu])  
Variable count not found!  
Error in:  
writetable(count',labr=tok[xu]) ...  
^
```

Fungsi `selecttable()` mengembalikan tabel baru dengan nilai dalam satu kolom yang dipilih dari vektor indeks. Pertama kita mencari indeks dari dua nilai kita di tabel token.

```
>v:=indexof(tok,["g","vg"])  
Variable or function tok not found.  
Error in:  
v:=indexof(tok,["g","vg"]) ...
```

Sekarang kita dapat memilih baris tabel, yang memiliki salah satu nilai v pada baris ke-5.

```
>MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5);
Variable or function MT not found.
Error in:
MT1:=MT[selectrows(MT, 5, v)]; i:=sortedrows(MT1, 5); ..
^
```

Sekarang kita dapat mencetak tabel, dengan nilai yang diekstraksi dan diurutkan di kolom ke-5.

```
>writetable(MT1[i],labc=hd,ctok=ctok,tok=tok,wc=7);
Variable or function i not found.
Error in:
writetable(MT1[i], labc=hd, ctok=ctok, tok=tok, wc=7); ..
^
```

Untuk statistik selanjutnya, kami ingin menghubungkan dua kolom tabel. Jadi kita ekstrak kolom 2 dan 4 dan urutkan tabelnya.

```
>i:=sortedrows(MT,[2,4]);...
> writetable(tablecol(MT[i],[2,4])',ctok=[1,2],tok=tok)
Variable or function MT not found.
Error in:
i:=sortedrows(MT, [2, 4]);      writetable(tablecol(MT[i],
^
```

Dengan `getstatistics()`, kita juga bisa menghubungkan jumlah dalam dua kolom tabel satu sama lain.

```
>MT24=tablecol(MT,[2,4]);...
> {xu1,xu2,count}=getstatistics(MT24[1],MT24[2]);...
> writetable(count,labr=tok[xu1],labc=tok[xu2])
Variable or function MT not found.
Error in:
MT24=tablecol(MT, [2, 4]); {xu1,xu2,count}=getstatistic
^
```

Sebuah tabel dapat ditulis ke file.

```
>filename="test.dat";...
> writetable(count,labr=tok[xu1],labc=tok[xu2],file=filename);
```

Variable or function count not found.

Error in:

```
filename="test.dat"; writetable(count,labr=tok[xul],labc=t  
^
```

Kemudian kita bisa membaca tabel dari file tersebut.

```
>{MT2,hd,tok2,hdr}=readtable(filename,>clabs,>rlabs); ...  
> writetable(MT2,labr=hdr,labc=hd)
```

Function readtable not found.

Try list ... to find functions!

Error in:

```
... T2,hd,tok2,hdr}=readtable(filename,&gt;clabs,&gt;rlabs  
^
```

Dan hapus file tersebut.

```
>fileremove(filename);
```

## DISTRIBUSI

Dengan `plot2d`, ada metode yang sangat mudah untuk memplot sebaran data eksperimen.

```
>p=normal(1,1000); //1000 random normal-distributed sample  
p
```

```
>plot2d(p,distribution=20,style="\f"); // plot the random  
sample p
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d(p,distribution=20,style="\f"); // plot the random  
^
```

```
>plot2d("qnormal(x,0,1)",add=1): // add the standard normal  
distribution plot
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("qnormal(x,0,1)",add=1): // add the standard normal  
^
```

Perlu diperhatikan perbedaan antara bar plot (sampel) dan kurva normal (distribusi sebenarnya). Masukkan kembali ketiga perintah untuk melihat hasil pengambilan sampel lainnya.

Berikut adalah perbandingan 10 simulasi dari 1000 nilai terdistribusi normal menggunakan apa yang disebut plot kotak. Plot ini menunjukkan median, kuartil 25% dan 75%, nilai minimal dan maksimal, serta outlier.

```
>p=normal(10,1000); boxplot(p):
```

```
Function boxplot not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
p=normal(10,1000); boxplot(p): ...  
^
```

Untuk menghasilkan bilangan bulat acak, Euler memiliki

intrandom. Mari kita simulasikan lemparan dadu dan plot distribusinya.

Kita menggunakan fungsi `getmultiplicities(v,x)`, yang menghitung seberapa sering elemen `v` muncul di `x`. Kemudian kita plot hasilnya menggunakan `kolomplot()`.

```
>k=intrandom(1,6000,6); ...  
> columnsplot(getmultiplicities(1:6,k)); ...  
> ygrid(1000,color=red):  
Function getmultiplicities not found.  
Try list ... to find functions!  
Error in:  
... (1,6000,6); columnsplot(getmultiplicities(1:6,k)); y
```

Meskipun `intrandom(n,m,k)` mengembalikan bilangan bulat yang terdistribusi secara seragam dari 1 hingga `k`, distribusi bilangan bulat lainnya dapat digunakan dengan `randpint()`.

Dalam contoh berikut, probabilitas untuk 1,2,3 masing-masing adalah 0,4,0,1,0,5.

```
>randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,%)  
Function randpint not found.  
Try list ... to find functions!  
Error in:  
randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,%)
```

Euler dapat menghasilkan nilai acak dari lebih banyak distribusi. Lihat referensinya.

Misalnya, kita mencoba distribusi eksponensial. Variabel acak kontinu `X` dikatakan berdistribusi eksponensial, jika PDF-nya diberikan oleh

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

dengan parameter

```
>plot2d(randexponential(1,1000,2),>distribution): ...  
Function randexponential not found.  
Try list ... to find functions!  
Error in:  
plot2d(randexponential(1,1000,2),>distribution): ...
```

Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan inversnya.

```
>plot2d("normaldis",-4,4):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("normaldis",-4,4): ...
```

Berikut ini adalah salah satu cara untuk memplot kuantil.

```
>plot2d("qnormal(x,1,1.5)",-4,6); ...
> plot2d("qnormal(x,1,1.5)",a=2,b=5,>add,>filled):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("qnormal(x,1,1.5)",-4,6); plot2d("qnormal(x,1,
```

Peluang berada di kawasan hijau adalah sebagai berikut.

```
>normaldis(5,1,1.5)-normaldis(2,1,1.5)
Built-in function normaldis needs 1 argument (got 3)!
Error in:
normaldis(5,1,1.5)-normaldis(2,1,1.5) ...
```

Ini dapat dihitung secara numerik dengan integral berikut.

```
>gauss("qnormal(x,1,1.5)",2,5)
Function gauss not found.
Try list ... to find functions!
Error in:
gauss("qnormal(x,1,1.5)",2,5) ...
```

Mari kita bandingkan distribusi binomial dengan distribusi normal yang mean dan deviasinya sama. Fungsi invbindis() menyelesaikan interpolasi linier antara nilai integer.

```
>invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1000))
Function invbindis not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1  
^
```

Fungsi `qdis()` adalah kepadatan distribusi chi-kuadrat. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian kita mendapatkan plot semua distribusi chi-kuadrat dengan derajat 5 sampai 30 dengan mudah dengan cara berikut.

```
>plot2d("qchidis(x,(5:5:50)')",0,50):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("qchidis(x,(5:5:50)')",0,50): ...  
^
```

Euler memiliki fungsi akurat untuk mengevaluasi distribusi. Mari kita periksa `chidis()` dengan integral.

Penamaannya mencoba untuk konsisten. Misalnya.,

- distribusi chi-kuadratnya adalah `chidis()`,
- fungsi kebalikannya adalah `invchidis()`,
- kepadatannya adalah `qchidis()`.

Pelengkap distribusi (ekor atas) adalah `chicdis()`.

```
>chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)
```

```
0.527633
```

```
Function integrate not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
chidis(1.5,2), integrate("qchidis(x,2)",0,1.5) ...  
^
```



## DISTRIBUSI DISKRIT

Untuk menentukan distribusi diskrit Anda sendiri, Anda dapat menggunakan metode berikut.

Pertama kita atur fungsi distribusinya.

```
>wd = 0|((1:6)+[-0.01,0.01,0,0,0,0])/6  
[0, 0.165, 0.335, 0.5, 0.666667, 0.833333, 1]
```

Artinya dengan probabilitas  $wd[i+1]-wd[i]$  kita menghasilkan nilai acak  $i$ .

Ini hampir merupakan distribusi yang seragam. Mari kita tentukan generator nomor acak untuk ini. Fungsi `find(v,x)` mencari nilai  $x$  pada vektor  $v$ . Fungsi ini juga berfungsi untuk vektor  $x$ .

```
>function wrongdice (n,m) := find(wd,random(n,m))
```

Kesalahannya sangat halus sehingga kita hanya melihatnya dengan banyak iterasi.

```
>columnplot(getmultiplicities(1:6,wrongdice(1,1000000))):  
Function getmultiplicities not found.  
Try list ... to find functions!  
Error in:  
... splot(getmultiplicities(1:6,wrongdice(1,1000000))
```

Berikut adalah fungsi sederhana untuk memeriksa keseragaman distribusi nilai  $1 \dots K$  dalam  $v$ . Kita menerima hasilnya, jika untuk semua frekuensi

$$\text{latex : } |f_i - 1/K| < \frac{\delta}{\sqrt{n}}.$$

```
>function checkrandom (v, delta=1) ...  
K=max(v) ; n=cols(v) ;  
fr=getfrequencies(v,1:K) ;  
return max(fr/n-1/K) < delta/sqrt(n) ;  
endfunction
```

Memang fungsinya menolak distribusi seragam.

```
>checkrandom(wrongdice(1,1000000))
```

```
Function getfrequencies not found.
Try list ... to find functions!
Try "trace errors" to inspect local variables after errors
checkrandom:
```

```
fr=getfrequencies(v,1:K);
```

Dan ia menerima generator acak bawaan.

```
>checkrandom(intrandom(1,1000000,6))
```

```
Function getfrequencies not found.
Try list ... to find functions!
Try "trace errors" to inspect local variables after errors
checkrandom:
```

```
fr=getfrequencies(v,1:K);
```

Kita dapat menghitung distribusi binomial. Pertama ada `binomialsun()`, yang mengembalikan probabilitas  $i$  atau kurang hit dari  $n$  percobaan.

```
>bindis(410,1000,0.4)
```

```
0.751401
```

Fungsi Beta terbalik digunakan untuk menghitung interval kepercayaan Clopper-Pearson untuk parameter  $p$ . Tingkat defaultnya adalah alfa.

Arti dari interval ini adalah jika  $p$  berada di luar interval, hasil pengamatan 410 dalam 1000 jarang terjadi.

```
>clopperpearson(410,1000)
```

```
Function clopperpearson not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
clopperpearson(410,1000) ...
```

Perintah berikut adalah cara langsung untuk mendapatkan hasil di atas. Namun untuk  $n$  yang besar, penjumlahan langsungnya tidak akurat dan lambat.

```
>p=0.4; i=0:410; n=1000; sum(bin(n,i)*pi*(1-p)(n-i))
```

```
0.751401
```

Omong-omong, `invbinsum()` menghitung kebalikan dari `binomialsun()`.

```
>invbindis(0.75,1000,0.4)
Function invbindis not found.
Try list ... to find functions!
Error in:
invbindis(0.75,1000,0.4) ...
^
```

Di Bridge, kami mengasumsikan 5 kartu beredar (dari 52) di dua tangan (26 kartu). Mari kita hitung probabilitas distribusi yang lebih buruk dari 3:2 (misalnya 0:5, 1:4, 4:1, atau 5:0).

```
>2*hypergeomsum(1,5,13,26)
Function hypergeomsum not found.
Try list ... to find functions!
Error in:
2*hypergeomsum(1,5,13,26) ...
^
```

Ada juga simulasi distribusi multinomial.

```
>randmultinomial(10,1000,[0.4,0.1,0.5])
Function randmultinomial not found.
Try list ... to find functions!
Error in:
randmultinomial(10,1000,[0.4,0.1,0.5]) ...
^
```



## MERENCANAKAN DATA

Untuk memetakan data, kami mencoba hasil pemilu Jerman sejak tahun 1990, diukur dalam jumlah kursi.

```
>BW := [ ...  
> 1990,662,319,239,79,8,17; ...  
> 1994,672,294,252,47,49,30; ...  
> 1998,669,245,298,43,47,36; ...  
> 2002,603,248,251,47,55,2; ...  
> 2005,614,226,222,61,51,54; ...  
> 2009,622,239,146,93,68,76; ...  
> 2013,631,311,193,0,63,64];
```

Untuk pesta, kami menggunakan rangkaian nama.

```
>P:=[“CDU/CSU”,“SPD”,“FDP”,“Gr”,“Li”];
```

Mari kita cetak persentasenya dengan baik.

Pertama kita mengekstrak kolom yang diperlukan. Kolom 3 sampai 7 adalah kursi masing-masing partai, dan kolom 2 adalah jumlah kursi seluruhnya. kolom adalah tahun pemilihan.

```
>BT:=BW[,3:7]; BT:=BT/sum(BT); YT:=BW[,1]’;
```

Kemudian statistiknya kita cetak dalam bentuk tabel. Kami menggunakan nama sebagai header kolom, dan tahun sebagai header untuk baris. Lebar default untuk kolom adalah wc=10, tetapi kami lebih memilih keluaran yang lebih padat. Kolom akan diperluas untuk label kolom, jika perlu.

```
>writetable(BT*100,wc=6,dc=0,>fixed,labc=P,labr=YT)
```

```
Function writetable not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... ritetable(BT*100,wc=6,dc=0,>fixed,labc=P,labr=YT)
```

Perkalian matriks berikut ini menjumlahkan persentase dua partai besar yang menunjukkan bahwa partai-partai kecil berhasil memperoleh suara di parlemen hingga tahun 2009.

```
>BT1:=(BT.[1;1;0;0;0])'*100
```

```
[84.29, 81.25, 81.1659, 82.7529, 72.9642, 61.8971, 7
```

Ada juga plot statistik sederhana. Kami menggunakannya untuk menampilkan garis dan titik secara bersamaan. Alternatifnya adalah memanggil plot2d dua kali dengan >add.

```
>statplot(YT,BT1,"b");
```

```
Function statplot not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
statplot(YT,BT1,"b"): ...  
^
```

Tentukan beberapa warna untuk setiap pesta.

```
>CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

```
Function rgb not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)]; ...  
^
```

Sekarang kita bisa memplot hasil pemilu 2009 dan perubahannya menjadi satu plot dengan menggunakan gambar. Kita dapat menambahkan vektor kolom ke setiap plot.

```
>figure(2,1);...
```

```
> figure(1); columnsplot(BW[6,3:7],P,color=CP); ...
```

```
> figure(2); columnsplot(BW[6,3:7]-BW[5,3:7],P,color=CP); ...
```

```
> figure(0):
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
figure(2,1); figure(1); columnsplot(BW[6,3:7],P,color=CP)  
^
```

Plot data menggabungkan deretan data statistik dalam satu plot.

```
>J:=BW[1,]'; DP:=BW[3:7]';...
```

```
> dataplot(YT,BT',color=CP);...
```

```
> labelbox(P,colors=CP,styles="[]",>points,w=0.2,x=0.3,y=0.4):
```

```
Variable CP not found!
```

Error in:

```
... =BW[,1]'; DP:=BW[,3:7]'; dataplot(YT,BT',color=CP
```

Plot kolom 3D memperlihatkan baris data statistik dalam bentuk kolom. Kami memberikan label untuk baris dan kolom. sudut adalah sudut pandang.

```
>columnplot3d(BT,scols=P,srows=YT,...  
> angle=30°,ccols=CP):
```

Variable CP not found!

Error in:

```
... splot3d(BT,scols=P,srows=YT, angle=30°,ccols=CP
```

Representasi lainnya adalah plot mosaik. Perhatikan bahwa kolom plot mewakili kolom matriks di sini. Karena panjang label CDU/CSU, kami mengambil jendela yang lebih kecil dari biasanya.

```
>shrinkwindow(>smaller);...  
> mosaicplot(BT',srows=YT,scols=P,color=CP,style="#");...  
> shrinkwindow():
```

Function shrinkwindow not found.

Try list ... to find functions!

Error in:

```
shrinkwindow(&gt;smaller); mosaicplot(BT',srows=YT,s
```

Kita juga bisa membuat diagram lingkaran. Karena hitam dan kuning membentuk koalisi, kami menyusun ulang elemen-elemennya.

```
>i=[1,3,5,4,2]; piechart(BW[6,3:7][i],color=CP[i],lab=P[i]):
```

CP is not a variable!

Error in:

```
i=[1,3,5,4,2]; piechart(BW[6,3:7][i],color=CP[i],lab=
```

Ini adalah jenis plot lainnya.

```
>starplot(normal(1,10)+4,lab=1:10,>rays):
```

Function starplot not found.

Try list ... to find functions!

Error in:

```
starplot(normal(1,10)+4,lab=1:10,&gt;rays): ...
```

Beberapa plot di plot2d bagus untuk statika. Berikut adalah plot impuls dari data acak, terdistribusi secara seragam di [0,1].

```
>plot2d(makeimpulse(1:10,random(1,10)),>bar):
Function makeimpulse not found.
Try list ... to find functions!
Error in:
plot2d(makeimpulse(1:10,random(1,10)),&gt;bar): ...
```

Namun untuk data yang terdistribusi secara eksponensial, kita mungkin memerlukan plot logaritmik.

```
>logimpulseplot(1:10,-log(random(1,10))*10):
Function logimpulseplot not found.
Try list ... to find functions!
Error in:
logimpulseplot(1:10,-log(random(1,10))*10): ...
```

Fungsi Columnplot() lebih mudah digunakan, karena hanya memerlukan vektor nilai. Selain itu, ia dapat mengatur labelnya ke apa pun yang kita inginkan, kami telah mendemonstrasikannya di tutorial ini.

Ini adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah kalimat dan membuat statistik.

```
>v=strtochar("the quick brown fox jumps over the lazy dog");
...
> w=ascii("a"):ascii("z"); x=getmultiplicities(w,v); ...
> cw=[]; for k=w; cw=cw|char(k); end; ...
> columnsplot(x,lab=cw,width=0.05):
Function getmultiplicities not found.
Try list ... to find functions!
Error in:
... w=ascii("a"):ascii("z"); x=getmultiplicities(w,v); cw=
```

Dimungkinkan juga untuk mengatur sumbu secara manual.



```

>n=10; p=0.4; i=0:n; x=bin(n,i)*pi*(1-p)(n-i); ...
> columnsplot(x,lab=i,width=0.05,<frame,<grid); ...
> yaxis(0,0:0.1:1,style="->",>left); axis(0,style="."); ...
> label("p",0,0.25), label("i",11,0); ...
> textbox(["Binomial distribution","with p=0.4"]):
Function columnsplot not found.
Try list ... to find functions!
Error in:
... -i); columnsplot(x,lab=i,width=0.05,&lt;frame,&lt;

```

Berikut ini cara memplot frekuensi bilangan dalam suatu vektor.

Kami membuat vektor bilangan acak bilangan bulat 1 hingga 6.

```
>v:=inrandom(1,10,10)
```

```
[8, 10, 6, 1, 7, 2, 2, 6, 9, 5]
```

Kemudian ekstrak nomor unik di v.

```
>vu:=unique(v)
```

```
[1, 2, 5, 6, 7, 8, 9, 10]
```

Dan plot frekuensi dalam plot kolom.

```
>columnsplot(getmultiplicities(vu,v),lab=vu,style="/"):
```

```
Function getmultiplicities not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
columnsplot(getmultiplicities(vu,v),lab=vu,style="/")
^
```

Kami ingin mendemonstrasikan fungsi distribusi nilai empiris.

```
>x=normal(1,20);
```

Fungsi empdist(x,vs) memerlukan array nilai yang diurutkan.

Jadi kita harus mengurutkan x sebelum kita dapat menggunakannya.

```
>xs=sort(x);
```

Kemudian kita plot distribusi empiris dan beberapa batang kepadatan ke dalam satu plot. Alih-alih plot batang untuk distribusi kali ini kami menggunakan plot gigi gergaji.

```
>figure(2,1); ...
```

```
> figure(1); plot2d("empdist",-4,4;xs); ...
```

```
> figure(2); plot2d(histo(x,v=-4:0.2:4,<bar)); ...
```

```
> figure(0):
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
figure(2,1); figure(1); plot2d("empdist",-4,4;xs); figure(0);
```

Plot sebar mudah dilakukan di Euler dengan plot titik biasa. Grafik berikut menunjukkan bahwa X dan X+Y jelas berkorelasi positif.

```
>x=normal(1,100); plot2d(x,x+rotright(x),>points,style=".."):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... ,100); plot2d(x,x+rotright(x),>points,style=".."):
```

Seringkali kita ingin membandingkan dua sampel dengan distribusi yang berbeda. Hal ini dapat dilakukan dengan plot kuantil-kuantil.

Untuk pengujiannya, kami mencoba distribusi student-t dan distribusi eksponensial.

```
>x=randt(1,1000,5); y=randnormal(1,1000,mean(x),dev(x));
```

```
...
```

```
> plot2d("x",r=6,style="-",yl="normal",xl="student-t",>vertical);
```

```
...
```

```
> plot2d(sort(x),sort(y),>points,color=red,style="x",>add):
```

```
Function randt not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
x=randt(1,1000,5); y=randnormal(1,1000,mean(x),dev(x)); pl
```

Plot tersebut dengan jelas menunjukkan bahwa nilai terdistribusi normal cenderung lebih kecil di ujung ekstrim.

Jika kita mempunyai dua distribusi yang ukurannya berbeda, kita dapat memperluas distribusi yang lebih kecil atau mengecilkan distribusi yang lebih besar. Fungsi berikut ini baik untuk keduanya. Dibutuhkan nilai median dengan persentase antara 0 dan 1.

```
>function medianexpand (x,n) := median(x,p=linspace(0,1,n-1));
```

Mari kita bandingkan dua distribusi yang sama.

```
>x=random(1000); y=random(400); ...  
> plot2d("x",0,1,style="--"); ...  
> plot2d(sort(medianexpand(x,400)),sort(y),>points,color=red,style="x",>add,  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
... (1000); y=random(400); plot2d("x",0,1,style="--")
```



## REGRESI DAN KORELASI

Regresi linier dapat dilakukan dengan fungsi `polyfit()` atau berbagai fungsi fit.

Sebagai permulaan kita menemukan garis regresi untuk data univariat dengan `polyfit(x,y,1)`.

```
>x=1:10; y=[2,3,1,5,6,3,7,8,9,8]; writetable(x'|y',labc=["x","y"])
Function writetable not found.
Try list ... to find functions!
Error in:
... ,5,6,3,7,8,9,8]; writetable(x'|y',labc=["x","y"])
```

Kami ingin membandingkan kecocokan yang tidak berbobot dan berbobot. Pertama koefisien kecocokan linier.

```
>p=polyfit(x,y,1)
Function polyfit not found.
Try list ... to find functions!
Error in:
p=polyfit(x,y,1) ...
^
```

Sekarang koefisien dengan bobot yang menekankan nilai terakhir.

```
>w &= "exp(-(x-10)^2/10)"; pw=polyfit(x,y,1,w=w(x))
Variable %e not found!
Use global variables or parameters for string evaluation.
Error in ^
Error in expression: %e^-((x-10)^2/10)
Error in:
w &= "exp(-(x-10)^2/10)"; pw=polyfit(x,y,1,w=w(x))
^
```

Kami memasukkan semuanya ke dalam satu plot untuk titik dan garis regresi, dan untuk bobot yang digunakan.

```
>figure(2,1); ...
```

```

> figure(1); statplot(x,y,"b",xl="Regression"); ...
> plot2d("evalpoly(x,p)",>add,color=blue,style="-"); ...
> plot2d("evalpoly(x,pw)",5,10,>add,color=red,style="-"); ...
> figure(2); plot2d(w,1,10,>filled,style="f",fillcolor=red,xl=w); ...
> figure(0):
Function figure not found.
Try list ... to find functions!
Error in:
figure(2,1);  figure(1); statplot(x,y,"b",xl="Regression")
^

```

Contoh lain kita membaca survei siswa, usia mereka, usia orang tua mereka dan jumlah saudara kandung dari sebuah file.

Tabel ini berisi “m” dan “f” di kolom kedua. Kami menggunakan variabel tok2 untuk mengatur terjemahan yang tepat alih-alih membiarkan readtable() mengumpulkan terjemahannya.

```

> {MS,hd}:=readtable("table1.dat",tok2:=["m","f"]); ...
> writetable(MS,labc=hd,tok2:=["m","f"]);
Function readtable not found.
Try list ... to find functions!
Error in:
{MS,hd}:=readtable("table1.dat",tok2:=["m","f"]);  writeta
^

```

Bagaimana usia bergantung satu sama lain? Kesan pertama muncul dari plot sebar berpasangan.

```

> scatterplots(tablecol(MS,3:5),hd[3:5]):
Variable or function MS not found.
Error in:
scatterplots(tablecol(MS,3:5),hd[3:5]): ...
^

```

Jelas terlihat bahwa usia ayah dan ibu saling bergantung satu sama lain. Mari kita tentukan dan plot garis regresinya.

```

> cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1)
MS is not a variable!
Error in:
cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1) ...
^

```

Ini jelas merupakan model yang salah. Garis regresinya adalah  $s=17+0,74t$ , dengan  $t$  adalah umur ibu dan  $s$  adalah umur ayah. Perbedaan usia mungkin sedikit bergantung pada usia, tapi tidak terlalu banyak.

Sebaliknya, kami mencurigai fungsi seperti  $s=a+t$ . Maka  $a$  adalah mean dari  $s-t$ . Ini adalah perbedaan usia rata-rata antara ayah dan ibu.

```
>da:=mean(cs[2]-cs[1])
cs is not a variable!
Error in:
da:=mean(cs[2]-cs[1]) ...
      ^
```

Mari kita plot ini menjadi satu plot sebar.

```
>plot2d(cs[1],cs[2],>points);...
> plot2d("evalpoly(x,ps)",color=red,style=".",>add);...
> plot2d("x+da",color=blue,>add):
cs is not a variable!
Error in:
plot2d(cs[1],cs[2],>points); plot2d("evalpoly(x,p
      ^
```

Berikut adalah plot kotak dari dua zaman tersebut. Ini hanya menunjukkan, bahwa usianya berbeda-beda.

```
>boxplot(cs,["mothers","fathers"]):
Variable or function cs not found.
Error in:
boxplot(cs,["mothers","fathers"]): ...
      ^
```

Menariknya, perbedaan median tidak sebesar perbedaan mean.

```
>median(cs[2])-median(cs[1])
cs is not a variable!
Error in:
median(cs[2])-median(cs[1]) ...
      ^
```

Koefisien korelasi menunjukkan korelasi positif.

```
>correl(cs[1],cs[2])
```

```
cs is not a variable!  
Error in:  
correl(cs[1],cs[2]) ...  
      ^
```

Korelasi pangkat merupakan ukuran keteraturan yang sama pada kedua vektor. Hal ini juga cukup positif.

```
>rankcorrel(cs[1],cs[2])  
cs is not a variable!  
Error in:  
rankcorrel(cs[1],cs[2]) ...  
      ^
```



## MEMBUAT FUNGSI BARU

Tentu saja, bahasa EMT dapat digunakan untuk memprogram fungsi-fungsi baru. Misalnya, kita mendefinisikan fungsi skewness.

dimana  $m$  adalah mean dari  $x$ .

```
>function skew (x:vector) ...  
m=mean(x) ;  
return sqrt (cols (x) ) *sum ( (x-m) ^3) / (sum ( (x-m) ^2) ) ^ (3/2)  
endfunction
```

Seperti yang Anda lihat, kita dapat dengan mudah menggunakan bahasa matriks untuk mendapatkan implementasi yang sangat singkat dan efisien. Mari kita coba fungsi ini.

```
>data=normal(20); skew(normal(10))  
Function mean not found.  
Try list ... to find functions!  
Try "trace errors" to inspect local variables after error.  
skew:  
m=mean(x) ;
```

Berikut adalah fungsi lainnya, yang disebut koefisien skewness Pearson.

```
>function skew1 (x) := 3*(mean(x)-median(x))/dev(x)  
>skew1(data)  
Function mean not found.  
Try list ... to find functions!  
Try "trace errors" to inspect local variables after error.  
skew1:  
useglobal; return 3*(mean(x)-median(x))/dev(x)  
Error in:  
skew1(data) ...  
^
```



## SIMULASI MONTE CARLO

Euler dapat digunakan untuk mensimulasikan kejadian acak. Kita telah melihat contoh sederhana di atas. Ini satu lagi, yang mensimulasikan 1000 kali lemparan 3 dadu, dan menanyakan pembagian jumlahnya.

```
>ds:=sum(intrandom(1000,3,6))'; fs=getmultiplicities(3:18,ds)
```

```
Function getmultiplicities not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... andom(1000,3,6))'; fs=getmultiplicities(3:18,ds)
```

Kita bisa merencanakannya sekarang.

```
>columnplot(fs,lab=3:18):
```

```
Variable or function fs not found.
```

```
Error in:
```

```
columnplot(fs,lab=3:18): ...  
^
```

Untuk menentukan distribusi yang diharapkan tidaklah mudah. Kami menggunakan rekursi tingkat lanjut untuk ini.

Fungsi berikut menghitung banyaknya cara bilangan k dapat direpresentasikan sebagai jumlah dari n bilangan dalam rentang 1 sampai m. Ia bekerja secara rekursif dengan cara yang jelas.

```
>function map countways (k; n, m) ...
```

```
if n==1 then return k>=1 && k<=m
```

```
else
```

```
sum=0;
```

```
loop 1 to m; sum=sum+countways(k-#,n-1,m); end;
```

```
return sum;
```

```
end;
```

```
endfunction
```

Berikut hasil pelemparan dadu sebanyak tiga kali.

```
>countways(5:25,5,5)
```

```
[1, 5, 15, 35, 70, 121, 185, 255, 320, 365,
```

```

255, 185, 121, 70, 35, 15, 5, 1]
>cw=countways(3:18,3,6)
[1, 3, 6, 10, 15, 21, 25, 27, 27, 25, 21, 15,
1]

```

Kami menambahkan nilai yang diharapkan ke plot.

```

>plot2d(cw/6^3*1000,>add); plot2d(cw/6^3*1000,>points,>add):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(cw/6^3*1000,&gt;add); plot2d(cw/6^3*1000,&gt;points
^

```

Untuk simulasi lain, deviasi nilai rata-rata n 0-1-variabel acak terdistribusi normal adalah  $1/\sqrt{n}$ .

```

>longformat; 1/sqrt(10)
Variable longformat not found!
Error in:
longformat; 1/sqrt(10) ...
^

```

Mari kita periksa ini dengan simulasi. Kami menghasilkan 10.000 kali 10 vektor acak.

```

>M=normal(10000,10); dev(mean(M)')
Function mean not found.
Try list ... to find functions!
Error in:
M=normal(10000,10); dev(mean(M)') ...
^

```

```

>plot2d(mean(M)',>distribution):
Function mean not found.
Try list ... to find functions!
Error in:
plot2d(mean(M)',&gt;distribution): ...
^

```

Median dari 10 bilangan acak berdistribusi normal 0-1 mempunyai deviasi yang lebih besar.

```

>dev(median(M)')

```

```
Function median not found.
Try list ... to find functions!
Error in:
dev(median(M)') ...
      ^
```

Karena kita dapat dengan mudah menghasilkan jalan acak, kita dapat mensimulasikan proses Wiener. Kami mengambil 1000 langkah dari 1000 proses. Kami kemudian memplot deviasi standar dan rata-rata langkah ke-n dari proses ini bersama dengan nilai yang diharapkan berwarna merah.

```
>n=1000; m=1000; M=cumsum(normal(n,m)/sqrt(m)); ...
> t=(1:n)/n; figure(2,1); ...
> figure(1); plot2d(t,mean(M')'); plot2d(t,0,color=red,>add); ...
> figure(2); plot2d(t,dev(M')'); plot2d(t,sqrt(t),color=red,>add); ...
> figure(0):
Function figure not found.
Try list ... to find functions!
Error in:
... msum(normal(n,m)/sqrt(m)); t=(1:n)/n; figure(2,1),
```



## TES

Tes adalah alat penting dalam statistik. Di Euler, banyak tes yang diterapkan. Semua pengujian ini mengembalikan kesalahan yang kita terima jika kita menolak hipotesis nol.

Misalnya, kami menguji lemparan dadu untuk distribusi yang seragam. Pada 600 kali lemparan, kami mendapatkan nilai berikut, yang kami masukkan ke dalam uji chi-kuadrat.

```
>chitest([90,103,114,101,103,89],dup(100,6)')
Function chitest not found.
Try list ... to find functions!
Error in:
chitest([90,103,114,101,103,89],dup(100,6)') ...
^
```

Uji chi-kuadrat juga memiliki mode yang menggunakan simulasi Monte Carlo untuk menguji statistiknya. Hasilnya seharusnya hampir sama. Parameter  $>p$  menafsirkan vektor  $y$  sebagai vektor probabilitas.

```
>chitest([90,103,114,101,103,89],dup(1/6,6)',>p,>montecarlo)
Function chitest not found.
Try list ... to find functions!
Error in:
... 0,103,114,101,103,89],dup(1/6,6)',&gt;p,&gt;monte
```

Kesalahan ini terlalu besar. Jadi kita tidak bisa menolak pemerataan. Ini tidak membuktikan bahwa dadu kita adil. Tapi kita tidak bisa menolak hipotesis kita.

Selanjutnya kita menghasilkan 1000 lemparan dadu menggunakan generator angka acak, dan melakukan tes yang sama.

```
>n=1000; t=random([1,n*6]); chitest(count(t*6,6),dup(n,6)')
Function chitest not found.
Try list ... to find functions!
Error in:
... =random([1,n*6]); chitest(count(t*6,6),dup(n,6)')
```

Mari kita uji nilai rata-rata 100 dengan uji-t.

```
>s=200+normal([1,100])*10; ...
> ttest(mean(s),dev(s),100,200)
Function mean not found.
Try list ... to find functions!
Error in:
s=200+normal([1,100])*10; ttest(mean(s),dev(s),100,200) . .
^
```

Fungsi ttest() memerlukan nilai mean, deviasi, jumlah data, dan nilai mean yang akan diuji.

Sekarang mari kita periksa dua pengukuran untuk mean yang sama. Kami menolak hipotesis bahwa keduanya mempunyai mean yang sama, jika hasilnya  $<0,05$ .

```
>tcomparedata(normal(1,10),normal(1,10))
Function tcomparedata not found.
Try list ... to find functions!
Error in:
tcomparedata(normal(1,10),normal(1,10)) ...
^
```

Jika kita menambahkan bias pada satu distribusi, kita akan mendapatkan lebih banyak penolakan. Ulangi simulasi ini beberapa kali untuk melihat efeknya.

```
>tcomparedata(normal(1,10),normal(1,10)+2)
Function tcomparedata not found.
Try list ... to find functions!
Error in:
tcomparedata(normal(1,10),normal(1,10)+2) ...
^
```

Pada contoh berikutnya, kita membuat 20 lemparan dadu acak sebanyak 100 kali dan menghitung yang ada di dalamnya. Rata-rata harus ada  $20/6=3,3$ .

```
>R=random(100,20); R=sum(R*6<=1)'; mean(R)
Function mean not found.
Try list ... to find functions!
```



```
Error in:
R=random(100,20); R=sum(R*6<=1)'; mean(R) ...
^
```

Sekarang kita bandingkan jumlah satuan dengan distribusi binomial. Pertama kita plot distribusinya.

```
>plot2d(R,distribution=max(R)+1,even=1,style="\"):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(R,distribution=max(R)+1,even=1,style="\"): ..
^
```

```
>t=count(R,21);
```

Kemudian kami menghitung nilai yang diharapkan.

```
>n=0:20; b=bin(20,n)*(1/6)n*(5/6)(20-n)*100;
```

Kita harus mengumpulkan beberapa angka untuk mendapatkan kategori yang cukup besar.

```
>t1=sum(t[1:2])|t[3:7]|sum(t[8:21]); ...
> b1=sum(b[1:2])|b[3:7]|sum(b[8:21]);
```

Uji chi-square menolak hipotesis bahwa distribusi kita merupakan distribusi binomial, jika hasilnya <0,05.

```
>chitest(t1,b1)
Function chitest not found.
Try list ... to find functions!
Error in:
chitest(t1,b1) ...
^
```

Contoh berikut berisi hasil dua kelompok orang (misalnya laki-laki dan perempuan) yang memilih satu dari enam partai.

```
>A=[23,37,43,52,64,74;27,39,41,49,63,76]; ...
> writetable(A,wc=6,labr=["m","f"],labc=1:6)
Function writetable not found.
Try list ... to find functions!
Error in:
... 6];      writetable(A,wc=6,labr=["m","f"],labc=1:6)
```

Kami ingin menguji independensi suara dari jenis kelamin. Uji tabel  $\chi^2$  melakukan hal ini. Dampaknya terlalu besar untuk menolak kemerdekaan. Jadi kita tidak bisa bilang, kalau voting tergantung jenis kelamin dari data tersebut.

```
>tabletest(A)
```

```
Function tabletest not found.  
Try list ... to find functions!  
Error in:  
tabletest(A) ...  
^
```

Berikut ini adalah tabel yang diharapkan, jika kita mengasumsikan frekuensi pemungutan suara yang diamati.

```
>writetable(expectedtable(A),wc=6,dc=1,labr=["m","f"],labc=1:6)
```

```
Function expectedtable not found.  
Try list ... to find functions!  
Error in:  
writetable(expectedtable(A),wc=6,dc=1,labr=["m","f"],labc=1:6)  
^
```

Kita dapat menghitung koefisien kontingensi yang dikoreksi. Karena sangat mendekati 0, kami menyimpulkan bahwa pemungutan suara tidak bergantung pada jenis kelamin.

```
>contingency(A)
```

```
Function contingency not found.  
Try list ... to find functions!  
Error in:  
contingency(A) ...  
^
```

## BEBERAPA TES LAGI

Selanjutnya kita menggunakan analisis varians (uji F) untuk menguji tiga sampel data yang berdistribusi normal untuk nilai mean yang sama. Metode tersebut disebut ANOVA (analisis varians). Di Euler, fungsi `varanalysis()` digunakan.

```
>x1=[109,111,98,119,91,118,109,99,115,109,94]; mean(x1),
Function mean not found.
Try list ... to find functions!
Error in:
... 09,111, 98,119, 91,118,109, 99,115,109, 94]; mean(x1)

>x2=[120,124,115,139,114,110,113,120,117]; mean(x2),
Function mean not found.
Try list ... to find functions!
Error in:
... 2=[120,124,115,139,114,110,113,120,117]; mean(x2)

>x3=[120,112,115,110,105,134,105,130,121,111]; mean(x3)
Function mean not found.
Try list ... to find functions!
Error in:
... 20,112,115,110,105,134,105,130,121,111]; mean(x3)

>varanalysis(x1,x2,x3)
Function varanalysis not found.
Try list ... to find functions!
Error in:
varanalysis(x1,x2,x3) ...
^
```

Artinya, kami menolak hipotesis nilai mean yang sama. Kami melakukan ini dengan probabilitas kesalahan 1,3%.

Ada juga uji median, yang menolak sampel data dengan distribusi rata-rata yang berbeda, menguji median dari sampel yang

disatukan.

```
>a=[56,66,68,49,61,53,45,58,54];  
>b=[72,81,51,73,69,78,59,67,65,71,68,71];  
>mediantest(a,b)
```

```
Function mediantest not found.  
Try list ... to find functions!  
Error in:  
mediantest(a,b) ...  
^
```

Tes kesetaraan lainnya adalah tes peringkat. Ini jauh lebih tajam daripada tes median.

```
>ranktest(a,b)
```

```
Function ranktest not found.  
Try list ... to find functions!  
Error in:  
ranktest(a,b) ...  
^
```

Pada contoh berikut, kedua distribusi mempunyai mean yang sama.

```
>ranktest(random(1,100),random(1,50)*3-1)
```

```
Function ranktest not found.  
Try list ... to find functions!  
Error in:  
ranktest(random(1,100),random(1,50)*3-1) ...  
^
```

Sekarang mari kita coba mensimulasikan dua perlakuan a dan b yang diterapkan pada orang yang berbeda.

```
>a=[8.0,7.4,5.9,9.4,8.6,8.2,7.6,8.1,6.2,8.9];  
>b=[6.8,7.1,6.8,8.3,7.9,7.2,7.4,6.8,6.8,8.1];
```

Tes signum memutuskan, apakah a lebih baik dari b.

```
>signtest(a,b)
```

```
Function signtest not found.  
Try list ... to find functions!  
Error in:  
signtest(a,b) ...
```

Ini kesalahan yang terlalu besar. Kita tidak dapat menolak bahwa a sama baiknya dengan b.

Uji Wilcoxon lebih tajam dibandingkan uji ini, namun mengandalkan nilai kuantitatif perbedaannya.

```
>wilcoxon(a,b)
Function wilcoxon not found.
Try list ... to find functions!
Error in:
wilcoxon(a,b) ...
^
```

Mari kita coba dua tes lagi menggunakan rangkaian yang dihasilkan.

```
>wilcoxon(normal(1,20),normal(1,20)-1)
Function wilcoxon not found.
Try list ... to find functions!
Error in:
wilcoxon(normal(1,20),normal(1,20)-1) ...
^
```

```
>wilcoxon(normal(1,20),normal(1,20))
Function wilcoxon not found.
Try list ... to find functions!
Error in:
wilcoxon(normal(1,20),normal(1,20)) ...
^
```



## ANGKA ACAK

Berikut ini adalah pengujian pembangkit bilangan acak. Euler menggunakan generator yang sangat bagus, jadi kita tidak perlu mengharapkan adanya masalah.

Pertama kita menghasilkan sepuluh juta angka acak di  $[0,1]$ .

```
>n:=10000000; r:=random(1,n);
```

Selanjutnya kita hitung jarak antara dua angka yang kurang dari 0,05.

```
>a:=0.05; d:=differences(nonzeros(r<a));
```

Function differences not found.

Try list ... to find functions!

Error in:

```
a:=0.05; d:=differences(nonzeros(r<a)); ...
```

Terakhir, kami memplot berapa kali, setiap jarak terjadi, dan membandingkannya dengan nilai yang diharapkan.

```
>m=getmultiplicities(1:100,d); plot2d(m); ...
```

```
> plot2d("n*(1-a)(x-1)*a2",color=red,>add):
```

Variable or function d not found.

Error in:

```
m=getmultiplicities(1:100,d); plot2d(m); plot2d("n*  
^
```

Hapus datanya.

```
>remvalue n;
```





## **PENGANTAR UNTUK PENGGUNA PROYEK R**

Jelasnya, EMT tidak bersaing dengan R sebagai paket statistik. Namun, ada banyak prosedur dan fungsi statistik yang tersedia di EMT juga. Jadi EMT dapat memenuhi kebutuhan dasar. Bagaimanapun, EMT hadir dengan paket numerik dan sistem aljabar komputer.

Notebook ini cocok untuk Anda yang sudah familiar dengan R, namun perlu mengetahui perbedaan sintaksis EMT dan R. Kami mencoba memberikan gambaran umum tentang hal-hal yang sudah jelas dan kurang jelas yang perlu Anda ketahui.

Selain itu, kami mencari cara untuk bertukar data antara kedua sistem.

Perhatikan bahwa ini masih dalam proses.



## SINTAKS DASAR

Hal pertama yang Anda pelajari di R adalah membuat vektor. Dalam EMT, perbedaan utamanya adalah operator : dapat mengambil ukuran langkah. Selain itu, ia mempunyai daya ikat yang rendah.

```
>n=10; 0:n/20:n-1
[0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5,
7, 7.5, 8, 8.5, 9]
```

Fungsi c() tidak ada. Dimungkinkan untuk menggunakan vektor untuk menggabungkan sesuatu.

Contoh berikut, seperti banyak contoh lainnya, berasal dari “Interduksi ke R” yang disertakan dengan proyek R. Jika Anda membaca PDF ini, Anda akan menemukan bahwa saya mengikuti jalurnya dalam tutorial ini.

```
>x=[10.4, 5.6, 3.1, 6.4, 21.7]; [x,0,x]
[10.4, 5.6, 3.1, 6.4, 21.7, 0, 10.4, 5.6, 3.1]
```

Operator titik dua dengan ukuran langkah EMT digantikan oleh fungsi seq() di R. Kita dapat menulis fungsi ini di EMT.

```
>function seq(a,b,c) := a:b:c; ...
> seq(0,-0.1,-1)
[0, -0.1, -0.2, -0.3, -0.4, -0.5, -0.6, -0.7,
```

Fungsi rep() dari R tidak ada di EMT. Untuk masukan vektor dapat dituliskan sebagai berikut.

```
>function rep(x:vector,n:index) := flatten(dup(x,n)); ...
> rep(x,2)
Function flatten not found.
Try list ... to find functions!
Try "trace errors" to inspect local variables after e
rep:
  useglobal; return flatten(dup(x,n))
Error in:
rep(x,2) ...
^
```

Perhatikan bahwa “=” atau “:=” digunakan untuk tugas. Operator “->” digunakan untuk satuan dalam EMT.

```
>125km -> " miles"
```

```
Variable km not found!
```

```
Error in:
```

```
125km ->" miles" ...  
^
```

Operator “<-” untuk penugasan memang menyesatkan, dan bukan ide yang baik untuk R. Berikut ini akan membandingkan a dan -4 di EMT.

```
>a=2; a<-4
```

```
0
```

Di R, “a<-4<3” berfungsi, tetapi “a<-4<-3” tidak. Saya juga memiliki ambiguitas serupa di EMT, tetapi saya mencoba menghilangkannya sedikit demi sedikit.

EMT dan R memiliki vektor bertipe boolean. Namun dalam EMT, angka 0 dan 1 digunakan untuk mewakili salah dan benar. Di R, nilai benar dan salah tetap bisa digunakan dalam aritmatika biasa seperti di EMT.

```
>x<5, %*x
```

```
[0, 0, 1, 0, 0]
```

```
[0, 0, 3.1, 0, 0]
```

EMT memunculkan kesalahan atau menghasilkan NAN tergantung pada tanda “kesalahan”.

```
>errors off; 0/0, isNAN(sqrt(-1)), errors on;
```

```
NAN
```

```
1
```

Stringnya sama di R dan EMT. Keduanya berada di lokal saat ini, bukan di Unicode.

Di R ada paket untuk Unicode. Di EMT, string dapat berupa string Unicode. String unicode dapat diterjemahkan ke pengkodean lokal dan sebaliknya. Selain itu, u”...” dapat berisi entitas HTML.

```
>u”© Ren&eacute; Grothmann”
```

```
© René Grothmann
```

Berikut ini mungkin atau mungkin tidak ditampilkan dengan

benar pada sistem Anda sebagai A dengan titik dan garis di atasnya. Itu tergantung pada font yang Anda gunakan.

```
>chartoutf([480])
```

Penggabungan string dilakukan dengan “+” atau “|”. Ini dapat mencakup angka, yang akan dicetak dalam format saat ini.

```
>“pi=”+pi
```

```
pi =      3.14159
```



## PENGINDEKSAN

Seringkali, ini akan berfungsi seperti di R.

Namun EMT akan menafsirkan indeks negatif dari belakang vektor, sedangkan R menafsirkan  $x[n]$  sebagai  $x$  tanpa elemen ke- $n$ .

```
>x, x[1:3], x[-2]
[10.4,  5.6,  3.1,  6.4,  21.7]
[10.4,  5.6,  3.1]
6.4
```

Perilaku R dapat dicapai dalam EMT dengan `drop()`.

```
>drop(x,2)
[10.4,  3.1,  6.4,  21.7]
```

Vektor logika tidak diperlakukan berbeda sebagai indeks di EMT, berbeda dengan R. Anda perlu mengekstrak elemen bukan nol terlebih dahulu di EMT.

```
>x, x>5, x[nonzeros(x>5)]
[10.4,  5.6,  3.1,  6.4,  21.7]
[1,  1,  0,  1,  1]
[10.4,  5.6,  6.4,  21.7]
```

Sama seperti di R, vektor indeks dapat berisi pengulangan.

```
[10.4,  5.6,  5.6,  10.4]
```

Namun penamaan indeks tidak dimungkinkan di EMT. Untuk paket statistik, hal ini sering kali diperlukan untuk memudahkan akses ke elemen vektor.

Untuk meniru perilaku ini, kita dapat mendefinisikan suatu fungsi sebagai berikut.

```
>function sel (v,i,s) := v[indexof(s,i)]; ...
> s=["first","second","third","fourth"]; sel(x,["first","third"],s)
[10.4,  3.1]
```





## TIPE DATA

EMT memiliki lebih banyak tipe data tetap daripada R. Jelasnya, di R terdapat vektor yang berkembang. Anda dapat mengatur vektor numerik kosong `v` dan memberikan nilai ke elemen `v[17]`. Hal ini tidak mungkin dilakukan di EMT.

Berikut ini agak tidak efisien.

```
>v=[]; for i=1 to 10000; v=v[i]; end;
```

EMT sekarang akan membuat vektor dengan `v` dan `i` ditambahkan pada tumpukan dan menyalin vektor tersebut kembali ke variabel global `v`.

Semakin efisien vektor telah ditentukan sebelumnya.

```
>v=zeros(10000); for i=1 to 10000; v[i]=i; end;
```

Untuk mengubah tipe tanggal di EMT, Anda dapat menggunakan fungsi seperti `kompleks()`.

```
>complex(1:4)
```

```
[ 1+0i, 2+0i, 3+0i, 4+0i ]
```

Konversi ke string hanya dimungkinkan untuk tipe data dasar. Format saat ini digunakan untuk penggabungan string sederhana. Tapi ada fungsi seperti `print()` atau `frac()`.

Untuk vektor, Anda dapat dengan mudah menulis fungsi Anda sendiri.

```
>function tostr(v)...
```

```
s="[";
```

```
loop 1 to length(v);
```

```
  s=s+print(v[#],2,0);
```

```
  if #<length(v) then s=s+", "; endif;
```

```
end;
```

```
return s+"]";
```

```
endfunction
```

```
>tostr(linspace(0,1,10))
```

```
Function length not found.
```

```
Try list ... to find functions!
```

Try "trace errors" to inspect local variables after errors  
tostr:

```
loop 1 to length(v);
```

Untuk komunikasi dengan Maxima, terdapat fungsi `convertmxm()`, yang juga dapat digunakan untuk memformat vektor untuk keluaran.

```
>convertmxm(1:10)
```

```
[1,2,3,4,5,6,7,8,9,10]
```

Untuk Latex perintah `tex` dapat digunakan untuk mendapatkan perintah Latex.

```
>tex(&[1,2,3])
```

```
\left[ 1 , 2 , 3 \right]
```

## FAKTOR DAN TABEL

Dalam pengantar R ada contoh yang disebut faktor.

Berikut ini adalah daftar wilayah 30 negara bagian.

```
>austates = ["tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa",  
...  
> "qld", "vic", "nsw", "vic", "qld", "qld", "sa", "tas", ...  
> "sa", "nt", "wa", "vic", "qld", "nsw", "nsw", "wa", ...  
> "sa", "act", "nsw", "vic", "vic", "act"];
```

Asumsikan, kita memiliki pendapatan yang sesuai di setiap negara bagian.

```
>incomes = [60, 49, 40, 61, 64, 60, 59, 54, 62, 69, 70, 42, 56,  
...  
> 61, 61, 61, 58, 51, 48, 65, 49, 49, 41, 48, 52, 46, ...  
> 59, 46, 58, 43];
```

Sekarang, kami ingin menghitung rata-rata pendapatan di suatu wilayah. Menjadi program statistik, R memiliki faktor() dan tapply() untuk ini.

EMT dapat melakukan hal ini dengan menemukan indeks wilayah dalam daftar wilayah unik.

```
>auterr=sort(unique(austates)); f=indexofsorted(auterr,austates)  
Need real matrix for unique  
Error in:  
auterr=sort(unique(austates)); f=indexofsorted(auterr,  
^
```

Pada titik itu, kita dapat menulis fungsi perulangan kita sendiri untuk melakukan sesuatu hanya untuk satu faktor.

Atau kita bisa meniru fungsi tapply() dengan cara berikut.

```
>function map tappl (i; f$:call, cat, x) ...  
u=sort(unique(cat));  
f=indexof(u,cat);  
return f$(x[nonzeros(f==indexof(u,i))]);  
endfunction
```

Ini agak tidak efisien, karena menghitung wilayah unik untuk setiap i, tetapi berhasil.

```
>tappl(auterr,"mean",austates,incomes)
```

```
Variable or function auterr not found.
```

```
Error in:
```

```
tappl(auterr, "mean", austates, incomes) ...  
^
```

Perhatikan bahwa ini berfungsi untuk setiap vektor wilayah.

```
>tappl(["act","nsw"],"mean",austates,incomes)
```

```
Need real matrix for unique
```

```
Try "trace errors" to inspect local variables after errors
```

```
tappl:
```

```
u=sort(unique(cat));
```

```
Error in map.
```

Sekarang, paket statistik EMT mendefinisikan tabel seperti di R. Fungsi `readtable()` dan `writetable()` dapat digunakan untuk input dan output.

Sehingga kita bisa mencetak rata-rata pendapatan negara di daerah secara bersahabat.

```
>writetable(tappl(auterr,"mean",austates,incomes),labc=auterr,wc=7)
```

```
Variable or function auterr not found.
```

```
Error in:
```

```
writetable(tappl(auterr, "mean", austates, incomes), labc=auterr,  
^
```

Kita juga bisa mencoba meniru perilaku R sepenuhnya.

Faktor-faktor tersebut harus disimpan dengan jelas dalam kumpulan beserta jenis dan kategorinya (negara bagian dan teritori dalam contoh kita). Untuk EMT, kami menambahkan indeks yang telah dihitung sebelumnya.

```
>function makef(t)...
```

```
## Factor data
```

```
## Returns a collection with data t, unique data, indices.
```

```
## See: tapply
```

```
u=sort(unique(t));
```

```
return ({t,u,indexofsorted(u,t)});
```

```
endfunction
```

```
>statef=makef(austates);
```

Need real matrix for unique

Try "trace errors" to inspect local variables after e.  
makef:

```
u=sort(unique(t));
```

Sekarang elemen ketiga dari koleksi akan berisi indeks.

```
>statef[3]
```

statef is not a variable!

Error in:

```
statef[3] ...  
^
```

Sekarang kita bisa meniru `tapply()` dengan cara berikut. Ini akan mengembalikan tabel sebagai kumpulan data tabel dan judul kolom.

```
>function tapply(t:vector,tf,f$:call)...
```

```
## Makes a table of data and factors
```

```
## tf : output of makef()
```

```
## See: makef
```

```
uf=tf[2]; f=tf[3]; x=zeros(length(uf));
```

```
for i=1 to length(uf);
```

```
    ind=nonzeros(f==i);
```

```
    if length(ind)==0 then x[i]=NAN;
```

```
    else x[i]=f$(t[ind]);
```

```
endif;
```

```
end;
```

```
return {{x,uf}};
```

```
endfunction
```

Kami tidak menambahkan banyak pengecekan tipe di sini. Satu-satunya tindakan pencegahan menyangkut kategori (faktor) yang tidak memiliki data. Tetapi kita harus memeriksa panjang `t` yang benar dan kebenaran pengumpulan `tf`.

Tabel ini dapat dicetak sebagai tabel dengan `writetable()`.

```
>writetable(tapply(incomes,statef,"mean"),wc=7)
```

Variable or function statef not found.

Error in:

```
writetable(tapply(incomes,statef,"mean"),wc=7) ...  
^
```

## ARRAY

EMT hanya memiliki dua dimensi untuk array. Tipe datanya disebut matriks. Namun, akan mudah untuk menulis fungsi untuk dimensi yang lebih tinggi atau perpustakaan C untuk ini.

R memiliki lebih dari dua dimensi. Di R array adalah vektor dengan bidang dimensi.

Dalam EMT, vektor adalah matriks dengan satu baris. Itu dapat dibuat menjadi matriks dengan `redim()`.

```
>shortformat; X=redim(1:20,4,5)
Variable shortformat not found!
Error in:
shortformat; X=redim(1:20,4,5) ...
      ^
```

Ekstraksi baris dan kolom, atau sub-matriks, mirip dengan R.

```
>X[,2:3]
X is not a variable!
Error in:
X[,2:3] ...
      ^
```

Namun, di R dimungkinkan untuk mengatur daftar indeks vektor tertentu ke suatu nilai. Hal yang sama mungkin terjadi di EMT hanya dengan satu putaran.

```
>function setmatrixvalue (M, i, j, v) ...
loop 1 to max(length(i), length(j), length(v))
  M[i{#}, j{#}] = v{#};
end;
endfunction
```

Kami mendemonstrasikan ini untuk menunjukkan bahwa matriks dilewatkan dengan referensi di EMT. Jika Anda tidak ingin mengubah matriks M asli, Anda perlu menyalinnya ke dalam fungsi.

```
>setmatrixvalue(X,1:3,3:-1:1,0); X,
Variable or function X not found.
```

```
Error in:
setmatrixvalue(X,1:3,3:-1:1,0); X, ...
^
```

Perkalian luar dalam EMT hanya dapat dilakukan antar vektor. Ini otomatis karena bahasa matriks. Satu vektor harus berupa vektor kolom dan vektor lainnya harus berupa vektor baris.

```
>(1:5)*(1:5)'
```

1	2	3	4
2	4	6	8
3	6	9	12
4	8	12	16
5	10	15	20

Dalam PDF pendahuluan untuk R terdapat contoh yang menghitung distribusi ab-cd untuk a,b,c,d yang dipilih dari 0 hingga n secara acak. Solusi dalam R adalah membentuk matriks 4 dimensi dan menjalankan table() di atasnya.

Tentu saja, hal ini dapat dicapai dengan satu putaran. Tapi loop tidak efektif di EMT atau R. Di EMT, kita bisa menulis loop di C dan itu akan menjadi solusi tercepat.

Namun kita ingin meniru perilaku R. Untuk melakukannya, kita perlu meratakan perkalian ab dan membuat matriks ab-cd.

```
>a=0:6; b=a'; p=flatten(a*b); q=flatten(p-p'); ...
> u=sort(unique(q)); f=getmultiplicities(u,q); ...
> statplot(u,f,"h"):
Function flatten not found.
Try list ... to find functions!
Error in:
a=0:6; b=a'; p=flatten(a*b); q=flatten(p-p'); u=sort(unique
^
```

Selain multiplisitas eksak, EMT dapat menghitung frekuensi dalam vektor.

```
>getfrequencies(q,-50:10:50)
Variable or function q not found.
Error in:
getfrequencies(q,-50:10:50) ...
^
```



Cara paling mudah untuk memplotnya sebagai distribusi adalah sebagai berikut.

```
>plot2d(q,distribution=11):  
Variable or function q not found.  
Error in:  
plot2d(q,distribution=11): ...  
^
```

Namun dimungkinkan juga untuk menghitung terlebih dahulu penghitungan dalam interval yang dipilih sebelumnya. Tentu saja, berikut ini menggunakan `getfrequencies()` secara internal.

Karena fungsi `histo()` mengembalikan frekuensi, kita perlu menskalakannya sehingga integral di bawah grafik batang adalah 1.

```
>{x,y}=histo(q,v=-55:10:55); y=y/sum(y)/differences(x); ...  
> plot2d(x,y,>bar,style="/"):  
Variable or function q not found.  
Error in:  
{x,y}=histo(q,v=-55:10:55); y=y/sum(y)/differences(x)  
^
```



## DAFTAR

EMT memiliki dua jenis daftar. Salah satunya adalah daftar global yang bisa berubah, dan yang lainnya adalah tipe daftar yang tidak bisa diubah. Kami tidak peduli dengan daftar global di sini.

Tipe daftar yang tidak dapat diubah disebut koleksi di EMT. Ini berperilaku seperti struktur di C, tetapi elemennya hanya diberi nomor dan tidak diberi nama.

$$>L=\{\text{"Fred"}, \text{"Flintstone"}, 40, [1990, 1992]\}$$

Fred

Flintstone

40

[1990, 1992]

Saat ini unsur-unsur tersebut tidak memiliki nama, meskipun nama dapat ditetapkan untuk tujuan khusus. Mereka diakses dengan nomor.

$$>(L[4])[2]$$

1992



## FILE INPUT DAN OUTPUT (MEMBACA DAN MENULIS DATA)

Anda sering kali ingin mengimpor matriks data dari sumber lain ke EMT. Tutorial ini memberi tahu Anda tentang banyak cara untuk mencapai hal ini. Fungsi sederhananya adalah `writematrix()` dan `readmatrix()`.

Mari kita tunjukkan cara membaca dan menulis vektor real ke sebuah file.

```
>a=random(1,100); mean(a), dev(a),  
Function mean not found.  
Try list ... to find functions!  
Error in:  
a=random(1,100); mean(a), dev(a), ...  
^
```

Untuk menulis data ke file, kita menggunakan fungsi `writematrix()`.

Karena pengenalan ini kemungkinan besar ada di direktori, di mana pengguna tidak memiliki akses tulis, kami menulis data ke direktori home pengguna. Untuk buku catatan sendiri, hal ini tidak diperlukan, karena file data akan ditulis ke dalam direktori yang sama.

```
>filename="test.dat";
```

Sekarang kita menulis vektor kolom `a'` ke file. Ini menghasilkan satu nomor di setiap baris file.

```
>writematrix(a',filename);  
Function writematrix not found.  
Try list ... to find functions!  
Error in:  
writematrix(a', filename); ...  
^
```

Untuk membaca data, kami menggunakan `readmatrix()`.

```
>a=readmatrix(filename)';  
Function readmatrix not found.  
Try list ... to find functions!
```

```
Error in:
a=readmatrix(filename)'; ...
      ^
```

Dan hapus file tersebut.

```
>fileremove(filename);
>mean(a), dev(a),
```

```
Function mean not found.
Try list ... to find functions!
Error in:
mean(a), dev(a), ...
      ^
```

Fungsi writematrix() atau writetable() dapat dikonfigurasi untuk bahasa lain.

Misalnya, jika Anda memiliki sistem Indonesia (titik desimal dengan koma), Excel Anda memerlukan nilai dengan koma desimal yang dipisahkan dengan titik koma dalam file csv (defaultnya adalah nilai yang dipisahkan koma). File berikut “test.csv” akan muncul di folder saat ini Anda.

```
>filename=“test.csv”;...
> writematrix(random(5,3),file=filename,separator=“,”);
Function writematrix not found.
Try list ... to find functions!
Error in:
... tematrix(random(5,3),file=filename,separator=","); ...
      ^
```

Anda sekarang dapat membuka file ini dengan Excel bahasa Indonesia secara langsung.

```
>fileremove(filename);
```

Terkadang kita memiliki string dengan token seperti berikut.

```
>s1:=“f m m f m m m f f f m m f”; ...
> s2:=“f f f m m f f”;
```

Untuk melakukan tokenisasi ini, kami mendefinisikan vektor token.

```
>tok:=[“f”,“m”]
```

f

m

Kemudian kita dapat menghitung berapa kali setiap token muncul dalam string, dan memasukkan hasilnya ke dalam tabel.

```
>M:=getmultiplicities(tok,strtokens(s1))_...
> getmultiplicities(tok,strtokens(s2));
Function getmultiplicities not found.
Try list ... to find functions!
Error in:
M:=getmultiplicities(tok,strtokens(s1))_      getmultipl
```

Tulis tabel dengan header token.

```
>writetable(M,labc=tok,labr=1:2,wc=8)
Function writetable not found.
Try list ... to find functions!
Error in:
writetable(M,labc=tok,labr=1:2,wc=8) ...
```

Untuk statika, EMT dapat membaca dan menulis tabel.

```
>file="test.dat"; open(file,"w"); ...
> writeln("A,B,C"); writematrix(random(3,3)); ...
> close();
Function writematrix not found.
Try list ... to find functions!
Error in:
... , "w"); writeln("A,B,C"); writematrix(random(3,3))
```

Filenya terlihat seperti ini.

```
>printfile(file)
Function printfile not found.
Try list ... to find functions!
Error in:
printfile(file) ...
```

Fungsi readtable() dalam bentuknya yang paling sederhana dapat membaca ini dan mengembalikan kumpulan nilai dan baris

judul.

```
>L=readtable(file,>list);
```

```
Function readtable not found.  
Try list ... to find functions!  
Error in:  
L=readtable(file,>list); ...  
^
```

Koleksi ini dapat dicetak dengan writetable() ke buku catatan, atau ke file.

```
>writetable(L,wc=10,dc=5)
```

```
Function writetable not found.  
Try list ... to find functions!  
Error in:  
writetable(L,wc=10,dc=5) ...  
^
```

Matriks nilai adalah elemen pertama dari L. Perhatikan bahwa mean() di EMT menghitung nilai rata-rata baris matriks.

```
>mean(L[1])
```

```
Function mean not found.  
Try list ... to find functions!  
Error in:  
mean(L[1]) ...  
^
```



## FILE CSV

Pertama, mari kita menulis matriks ke dalam file. Untuk outputnya, kami membuat file di direktori kerja saat ini.

```
>file="test.csv";...
> M=random(3,3); writematrix(M,file);
Function writematrix not found.
Try list ... to find functions!
Error in:
... e="test.csv"; M=random(3,3); writematrix(M,file)
```

Berikut isi file ini.

```
>printfile(file)
Function printfile not found.
Try list ... to find functions!
Error in:
printfile(file) ...
^
```

CSV ini dapat dibuka pada sistem berbahasa Inggris ke Excel dengan klik dua kali. Jika Anda mendapatkan file seperti itu di sistem Jerman, Anda perlu mengimpor data ke Excel dengan memperhatikan titik desimal.

Namun titik desimal juga merupakan format default untuk EMT. Anda dapat membaca matriks dari file dengan `readmatrix()`.

```
>readmatrix(file)
Function readmatrix not found.
Try list ... to find functions!
Error in:
readmatrix(file) ...
^
```

Dimungkinkan untuk menulis beberapa matriks ke satu file. Perintah `open()` dapat membuka file untuk ditulis dengan parameter “w”. Standarnya adalah “r” untuk membaca.

```

>open(file,"w"); writematrix(M); writematrix(M'); close();
Function writematrix not found.
Try list ... to find functions!
Error in:
open(file,"w"); writematrix(M); writematrix(M'); close();
^

```

Matriks dipisahkan oleh garis kosong. Untuk membaca matriks, buka file dan panggil readmatrix() beberapa kali.

```

>open(file); A=readmatrix(); B=readmatrix(); A==B, close();
Function readmatrix not found.
Try list ... to find functions!
Error in:
open(file); A=readmatrix(); B=readmatrix(); A==B, close();
^

```

Di Excel atau spreadsheet serupa, Anda dapat mengekspor matriks sebagai CSV (nilai yang dipisahkan koma). Di Excel 2007, gunakan “save as” dan “other format”, lalu pilih “CSV”. Pastikan tabel saat ini hanya berisi data yang ingin Anda ekspor.

Ini sebuah contoh.

```

>printfile("excel-data.csv")
Function printfile not found.
Try list ... to find functions!
Error in:
printfile("excel-data.csv") ...
^

```

Seperti yang Anda lihat, sistem bahasa Jerman saya menggunakan titik koma sebagai pemisah dan koma desimal. Anda dapat mengubahnya di pengaturan sistem atau di Excel, tetapi hal ini tidak diperlukan untuk membaca matriks menjadi EMT.

Cara termudah untuk membaca ini ke dalam Euler adalah readmatrix(). Semua koma diganti dengan titik dengan parameter >koma. Untuk CSV bahasa Inggris, hilangkan saja parameter ini.

```

>M=readmatrix("excel-data.csv",>comma)
Function readmatrix not found.
Try list ... to find functions!

```

```
Error in:
M=readmatrix("excel-data.csv", >comma) ...
      ^
```

Mari kita rencanakan ini.

```
>plot2d(M'[1],M'[2:3],>points,color=[red,green]'):
Variable red not found!
Error in:
plot2d(M'[1],M'[2:3],>points,color=[red,green]'):
      ^
```

Ada cara yang lebih mendasar untuk membaca data dari suatu file. Anda dapat membuka file dan membaca angka baris demi baris. Fungsi `getvectorline()` akan membaca angka dari sebaris data. Secara default, ini mengharapkan titik desimal. Tapi bisa juga menggunakan koma desimal, jika Anda memanggil `setdecimaldot(",")` sebelum Anda menggunakan fungsi ini.

Fungsi berikut adalah contohnya. Itu akan berhenti di akhir file atau baris kosong.

```
>function myload (file) ...
open(file);
M=[];
repeat
    until eof();
    v=getvectorline(3);
    if length(v)>0 then M=M_v; else break; endif;
end;
return M;
close(file);
endfunction

>myload(file)

Function length not found.
Try list ... to find functions!
Try "trace errors" to inspect local variables after e
myload:
    if length(v)>0 then M=M_v; else break; endif;
```

Dimungkinkan juga untuk membaca semua angka dalam file itu dengan `getvector()`.

```

>open(file); v=getvector(10000); close(); redim(v[1:9],3,3)
Index 1 out of bounds!
Error in:
... (file); v=getvector(10000); close(); redim(v[1:9],3,3)
^

```

Oleh karena itu sangat mudah untuk menyimpan suatu vektor nilai, satu nilai di setiap baris dan membaca kembali vektor ini.

```

>v=random(1000); mean(v)
Function mean not found.
Try list ... to find functions!
Error in:
v=random(1000); mean(v) ...
^

>writematrix(v',file); mean(readmatrix(file)')
Function writematrix not found.
Try list ... to find functions!
Error in:
writematrix(v',file); mean(readmatrix(file)') ...
^

```

## MENGGUNAKAN TABEL

Tabel dapat digunakan untuk membaca atau menulis data numerik. Misalnya, kita menulis tabel dengan header baris dan kolom ke sebuah file.

```
>file="test.tab"; M=random(3,3); ...
> open(file,"w"); ...
> writetable(M,separator="," ,labc=["one","two","three"]); ...
> close(); ...
> printfile(file)
Function writetable not found.
Try list ... to find functions!
Error in:
... table(M,separator="," ,labc=["one","two","three"])
```

Ini dapat diimpor ke Excel.

Untuk membaca file di EMT, kami menggunakan readtable().

```
>{M,headings}=readtable(file,>clabs); ...
> writetable(M,labc=headings)
Function readtable not found.
Try list ... to find functions!
Error in:
{M,headings}=readtable(file,>clabs); writetable(M,
```



## MENGANALISIS GARIS

Anda bahkan dapat mengevaluasi setiap baris dengan tangan. Misalkan, kita memiliki baris dengan format berikut.

```
>line="2020-11-03,Tue,1'114.05"
```

```
2020-11-03, Tue, 1' 114.05
```

Pertama, kita dapat memberi token pada garis tersebut.

```
>vt=strtokens(line)
```

```
2020-11-03
```

```
Tue
```

```
1' 114.05
```

Kemudian kita dapat mengevaluasi setiap elemen garis menggunakan evaluasi yang sesuai.

```
>day(vt[1]),...
```

```
>indexof(["mon","tue","wed","thu","fri","sat","sun"],tolower(vt[2])),
```

```
...
```

```
>strepl(vt[3],""),"")()
```

```
Function day not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
day(vt[1]),  indexof(["mon","tue","wed","thu","fri","sun"],
```

Dengan menggunakan ekspresi reguler, dimungkinkan untuk mengekstrak hampir semua informasi dari sebaris data.

Asumsikan kita memiliki baris berikut sebuah dokumen HTML.

```
>line="<tr><td>1145.45</td><td>5.6</td><td>-4.5</td><tr>"
```

```
&lt;tr>&lt;td>1145.45&lt;/td>&lt;td>5.6&lt;/td>&lt;td>-4.5&lt;/td>&lt;/tr>"
```

Untuk mengekstraknya, kami menggunakan ekspresi reguler, yang mencari

- tanda kurung tutup >,
- string apa pun yang tidak mengandung tanda kurung dengan sub-pencocokan "(...)",

- braket pembuka dan penutup menggunakan solusi terpendek,
- sekali lagi string apa pun yang tidak mengandung tanda kurung,
- dan tanda kurung buka <.

Ekspresi reguler agak sulit dipelajari tetapi sangat ampuh.

```
>{pos,s,vt}=strxfind(line,">([^\>]+)<.+?>([^\>]+)<");
```

Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk sub-kecocokan.

```
>for k=1:length(vt); vtk, end;
```

Function length not found.

Try list ... to find functions!

Error in:

```
for k=1:length(vt); vt[k](), end; ...
      ^
```

Berikut adalah fungsi yang membaca semua item numerik antara <td> dan </td>.

```
>function readtd (line) ...
```

```
v=[]; cp=0;
```

```
repeat
```

```
{pos,s,vt}=strxfind(line,"<td.*?>(.+?)</td>",cp);
```

```
until pos==0;
```

```
if length(vt)>0 then v=v|vt[1]; endif;
```

```
cp=pos+strlen(s);
```

```
end;
```

```
return v;
```

```
endfunction
```

```
>readtd(line+"<td>non-numerical</td>")
```

Function length not found.

Try list ... to find functions!

Try "trace errors" to inspect local variables after errors

```
readtd:
```

```
if length(vt)>0 then v=v|vt[1]; endif;
```



## MEMBACA DARI WEB

Situs web atau file dengan URL dapat dibuka di EMT dan dapat dibaca baris demi baris.

Dalam contoh, kita membaca versi terkini dari situs EMT. Kami menggunakan ekspresi reguler untuk memindai “Versi ...” dalam sebuah judul.

```
>function readversion () ...
urlopen("http://www.euler-math-toolbox.de/Programs/Cha
repeat
    until urleof();
    s=urlgetline();
    k=strfind(s,"Version ",1);
    if k>0 then substring(s,k,strfind(s,"<",k)-1), break
end;
urlclose();
endfunction

>readversion
Version 2024-01-12
```



## INPUT DAN OUTPUT VARIABEL

Anda dapat menulis variabel dalam bentuk definisi Euler ke file atau ke baris perintah.

```
>writevar(pi,"mypi");  
Function writevar not found.  
Try list ... to find functions!  
Error in:  
writevar(pi,"mypi"); ...  
      ^
```

Untuk pengujian, kami membuat file Euler di direktori kerja EMT.

```
>file="test.e";...  
> writevar(random(2,2),"M",file);...  
> printfile(file,3)  
Function writevar not found.  
Try list ... to find functions!  
Error in:  
file="test.e"; writevar(random(2,2),"M",file); printf...  
      ^
```

Sekarang kita dapat memuat file tersebut. Ini akan mendefinisikan matriks M.

```
>load(file); show M,  
Variable show not found!  
Error in:  
load(file); show M, ...  
      ^
```

Omong-omong, jika writevar() digunakan pada suatu variabel, definisi variabel dengan nama variabel tersebut akan dicetak.

```
>writevar(M); writevar(inch$)  
Function writevar not found.  
Try list ... to find functions!  
Error in:
```

```
writevar(M); writevar(inch$) ...
^
```

Kita juga bisa membuka file baru atau menambahkan file yang sudah ada. Dalam contoh kita menambahkan file yang dibuat sebelumnya.

```
>open(file,"a");...
> writevar(random(2,2),"M1");...
> writevar(random(3,1),"M2");...
> close();
Function writevar not found.
Try list ... to find functions!
Error in:
open(file,"a"); writevar(random(2,2),"M1"); writevar(rando
^
```

```
>load(file); show M1; show M2;
Variable show not found!
Error in:
load(file); show M1; show M2; ...
^
```

Untuk menghapus file apa pun, gunakan `fileremove()`.

```
>fileremove(file);
```

Vektor baris dalam suatu file tidak memerlukan koma, jika setiap angka berada pada baris baru. Mari kita buat file seperti itu, tulis setiap baris satu per satu dengan `writeln()`.

```
>open(file,"w"); writeln("M = ["); ...
> for i=1 to 5; writeln(" "+random()); end; ...
> writeln("];"); close(); ...
> printfile(file)
Function printfile not found.
Try list ... to find functions!
Error in:
... ()); end; writeln("];"); close(); printfile(file) ...
^
```

```
>load(file); M
[0.396904, 0.875874, 0.71702, 0.0416905, 0.299661]
```

**PLOT 2D**

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris. `unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip bookmark xurl`



## **NASYWA PLOT 2**

Nama : Nasywa Fierda Azizah

NIM : 23030630068

Prodi : Matematika B 23





## MENGGAMBAR GRAFIK 2D DENGAN EMT

Notebook ini menjelaskan tentang cara menggambar berbagai kurva dan grafik 2D dengan software EMT. EMT menyediakan fungsi `plot2d()` untuk menggambar berbagai kurva dan grafik dua dimensi (2D).

### Basic Plots

Ada fungsi plot yang sangat mendasar. Terdapat koordinat layar yang selalu berkisar antara 0 hingga 1024 di setiap sumbu, tidak peduli apakah layarnya berbentuk persegi atau tidak. Terdapat koordinat plot yang dapat diatur dengan `setplot()`. Pemetaan antar koordinat bergantung pada jendela plot saat ini. Misalnya, `shrinkwindow()` default memberikan ruang untuk label sumbu dan judul plot.

Macam-macam basic plot :

```
>clg; // untuk membersihkan layar
```

```
>window(0,0,1024,1024); // gunakan semua window
```

```
Built-in function window needs 0 arguments (got 4)!
```

```
Error in:
```

```
window(0,0,1024,1024); // gunakan semua window ...  
^
```

```
>setplot(0,1,0,1); // koordinat set plot
```

```
Built-in function setplot needs 1 argument (got 4)!
```

```
Error in:
```

```
setplot(0,1,0,1); // koordinat set plot ...  
^
```

```
>hold on; // untuk memulai overwrite mode
```

```
>n=100; X=random(n,2); Y=random(n,2); // untuk membuat  
koordinat acak
```

```
>colors=rgb(random(n),random(n),random(n)); // get random  
colors
```

```
Function rgb not found.
```

```
Try list ... to find functions!
```

Error in:

```
colors=rgb(random(n),random(n),random(n)); // get random c  
^
```

```
>loop 1 to n; color(colors[#]); plot(X[#],Y[#]); end; // plot
```

colors is not a variable!

Error in:

```
loop 1 to n; color(colors[#]); plot(X[#],Y[#]); end; // pl  
^
```

```
>hold off; // mengakhiri overwrite mode
```

```
>insimg; // memasukkan ke notebook notebook
```

Variable insimg not found!

Error in:

```
insimg; // memasukkan ke notebook notebook ...  
^
```

```
>reset;
```

Variable reset not found!

Error in:

```
reset; ...  
^
```

Grafik perlu ditahan, karena perintah plot() akan menghapus jendela plot.

Untuk menghapus semua yang kami lakukan, maka menggunakan reset().

Untuk menampilkan gambar hasil plot di layar notebook, perintah plot2d() dapat diakhiri dengan titik dua (:).

Cara lain, adalah perintah plot2d() diakhiri dengan titik koma (;), kemudian menggunakan perintah insimg() untuk menampilkan gambar hasil plot.

Contoh lain, kita menggambar plot sebagai sisipan di plot lain. Hal ini dilakukan dengan mendefinisikan jendela plot yang lebih kecil. Perhatikan bahwa jendela ini tidak memberikan ruang untuk label sumbu di luar jendela plot. Maka Kita harus menambahkan beberapa margin sesuai kebutuhan.

```
>plot2d("x^3-x");
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("x^3-x"); ...  
^
```

```
>xw=200; yw=100; ww=300; hw=300;
```

```
>ow=window();
```

```
>window(xw,yw,xw+ww,yw+hw);
```

Built-in function window needs 0 arguments (got 4)!

Error in:

```
window(xw,yw,xw+ww,yw+hw); ...  
^
```

```
>hold on;
```

```
>barclear(xw-50,yw-10,ww+60,ww+60);
```

Function barclear not found.

Try list ... to find functions!

Error in:

```
barclear(xw-50,yw-10,ww+60,ww+60); ...  
^
```

```
>plot2d("x^4-x",grid=6):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("x^4-x",grid=6): ...  
^
```

```
>hold off;
```

```
>window(ow);
```

Plot dengan banyak gambar dicapai dengan cara yang sama. Ada fungsi utility figure() untuk ini.

## Plot Aspect

Plot default menggunakan jendela plot persegi. Anda dapat mengubahnya dengan fungsi aspek(). Jangan lupa untuk mengatur ulang aspeknya nanti. Anda juga dapat mengubah default ini di menu dengan "Set Aspect" ke rasio aspek tertentu atau ke ukuran jendela grafik saat ini.

Tapi Anda juga bisa mengubahnya untuk satu plot. Untuk ini, ukuran area plot saat ini diubah, dan jendela diatur sehingga label memiliki cukup ruang.

```
>aspect(1); // rasio panjang dan lebar 2:1
```

```
>plot2d(["sin(x)","cos(x)"],0,2pi):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d(["sin(x)", "cos(x)"], 0, 2pi): ...  
^
```

```
>aspect(2); // rasio panjang dan lebar 2:1
```

```
>plot2d(["sin(x)","cos(x)"],0,2pi):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d(["sin(x)", "cos(x)"], 0, 2pi): ...  
^
```

```
>aspect(3); // rasio panjang dan lebar 2:1
```

```
>plot2d(["sin(x)","cos(x)"],0,2pi):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d(["sin(x)", "cos(x)"], 0, 2pi): ...  
^
```

```
>aspect();
```

```
Built-in function aspect needs 1 argument (got 0)!
```

```
Error in:
```

```
aspect(); ...  
^
```

```
>reset;
```

```
Variable reset not found!
```

```
Error in:
```

```
reset; ...  
^
```

Fungsi reset() mengembalikan default plot termasuk rasio aspek.

## PLOT 2D DI EULER

EMT Math Toolbox memiliki plot dalam 2D, baik untuk data maupun fungsi. EMT menggunakan fungsi `plot2d`. Fungsi ini dapat memplot fungsi dan data.

Dimungkinkan untuk membuat plot di Maxima menggunakan Gnuplot atau dengan Python menggunakan Math Plot Lib.

Euler dapat membuat plot 2D

- ekspresi
- fungsi, variabel, atau kurva berparameter,
- vektor nilai  $x$ - $y$ ,
- awan titik di pesawat,
- kurva implisit dengan level atau wilayah level.
- Fungsi kompleks

Gaya plot mencakup berbagai gaya untuk garis dan titik, plot batang, dan plot berbayang.



## PLOT EKSPRESI ATAU VARIABEL

Ekspresi tunggal dalam “x” (misalnya “ $4*x^2$ ”) atau nama suatu fungsi (misalnya “f”) menghasilkan grafik fungsi tersebut.

Berikut adalah contoh paling dasar, yang menggunakan rentang default dan menetapkan rentang y yang tepat agar sesuai dengan plot fungsinya.

Catatan: Jika Anda mengakhiri baris perintah dengan titik dua “:”, plot akan dimasukkan ke dalam jendela teks.

```
>plot2d("x^2");  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
plot2d("x^2"): ...  
^  
  
>aspect(1.5); plot2d("x^3-x");  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
aspect(1.5); plot2d("x^3-x"): ...  
^  
  
>a:=5.6; plot2d("exp(-a*x^2)/a"); insimg(30);  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
a:=5.6; plot2d("exp(-a*x^2)/a"); insimg(30); ...  
^
```

Dari beberapa contoh sebelumnya Anda dapat melihat bahwa aslinya gambar plot menggunakan sumbu X dengan rentang nilai dari -2 sampai dengan 2. Untuk mengubah rentang nilai X dan Y, Anda dapat menambahkan nilai-nilai batas X (dan Y) di belakang ekspresi yang digambar.

Rentang plot diatur dengan parameter yang ditetapkan sebagai berikut

- a,b: rentang x (default -2,2)
- c,d: rentang y (default: skala dengan nilai)
- r: alternatifnya radius di sekitar pusat plot
- cx,cy: koordinat pusat plot (default 0,0)

```
>plot2d("x^3-x",-1,2):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("x^3-x",-1,2): ...
      ^
```

```
>plot2d("sin(x)",-2*pi,2*pi):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("sin(x)",-2*pi,2*pi): ...
      ^
```

```
>plot2d("cos(x)","sin(3*x)",xmin=0,xmax=2pi):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("cos(x)","sin(3*x)",xmin=0,xmax=2pi): ...
      ^
```

Alternatif untuk titik dua adalah perintah `insimg(baris)`, yang menyisipkan plot yang menempati sejumlah baris teks tertentu.

Dalam opsi, plot dapat diatur agar muncul di jendela terpisah yang dapat diubah ukurannya.

Untuk membagi jendela menjadi beberapa plot, gunakan perintah `figure()`. Dalam contoh, kita memplot  $x^1$  hingga  $x^4$  menjadi 4 bagian jendela. `gambar(0)` mengatur ulang jendela default.

```
>reset;
```

Variable reset not found!

Error in:

```
reset; ...
      ^
```

```
>figure(2,2); ...
```



```
> for n=1 to 4; figure(n); plot2d("x^"+n); end; ...
```

```
> figure(0):
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
figure(2,2); for n=1 to 4; figure(n); plot2d("x^"+n);  
^
```

Di `plot2d()`, ada gaya alternatif yang tersedia dengan `grid=x`. Untuk gambaran umum, kami menampilkan berbagai gaya kisi dalam satu gambar (lihat di bawah untuk perintah `figure()`). Gaya `grid=0` tidak disertakan. Ini tidak menunjukkan kisi dan bingkai.

```
>figure(3,3); ...
```

```
> for k=1:9; figure(k); plot2d("x^3-x",-2,1,grid=k); end; ...
```

```
> figure(0):
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
figure(3,3); for k=1:9; figure(k); plot2d("x^3-x",-2,  
^
```

Jika argumen pada `plot2d()` adalah ekspresi yang diikuti oleh empat angka, angka-angka tersebut adalah rentang  $x$  dan  $y$  untuk plot tersebut.

Alternatifnya,  $a$ ,  $b$ ,  $c$ ,  $d$  dapat ditentukan sebagai parameter yang ditetapkan sebagai  $a$

Pada contoh berikut, kita mengubah gaya kisi, menambahkan label, dan menggunakan label vertikal untuk sumbu  $y$ .

```
>aspect(1.5); plot2d("sin(x)",0,2pi,-1.2,1.2,grid=3,xl="x",yl="sin(x)");
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... sin(x)",0,2pi,-1.2,1.2,grid=3,xl="x",yl="sin(x)");
```

```
>plot2d("sin(x)+cos(2*x)",0,4pi):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("sin(x)+cos(2*x)",0,4pi): ...
```

Fungsi atau ekspresi di plot2d dievaluasi secara adaptif. Agar lebih cepat, nonaktifkan plot adaptif dengan <adaptive dan tentukan jumlah subinterval dengan n=... Hal ini hanya diperlukan dalam kasus yang jarang terjadi.

```
>plot2d("sign(x)*exp(-x^2)",-1,1,<adaptive,n=10000):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
... lot2d("sign(x)*exp(-x^2)",-1,1,&lt;adaptive,n=10000):
```

```
>plot2d("x^x",r=1.2,cx=1,cy=1):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("x^x",r=1.2,cx=1,cy=1): ...
```

Parameter square=true (atau >square) memilih rentang y secara otomatis sehingga hasilnya adalah jendela plot persegi. Perhatikan bahwa secara default, Euler menggunakan spasi persegi di dalam jendela plot.

```
>plot2d("x^3-x",>square):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("x^3-x",&gt;square): ...
```

```
>plot2d(''integrate("sin(x)*exp(-x^2)",0,x)',0,2):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d(''integrate("sin(x)*exp(-x^2)",0,x)',0,2): ...
```

Jika Anda memerlukan lebih banyak ruang untuk label y, panggil shrinkwindow() dengan parameter lebih kecil, atau tetapkan

nilai positif untuk “smaller” di plot2d().

```
>plot2d(“gamma(x)”,1,10,yl=“y-values”,smaller=6,<vertical):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
... gamma(x) ", 1, 10, yl="y-values", smaller=6, &lt;vertical):
```

```
>x=linspace(0,2pi,1000); plot2d(sin(5x),cos(7x)):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
x=linspace(0,2pi,1000); plot2d(sin(5x),cos(7x)): ...  
^
```

```
>a:=5.6; expr &= exp(-a*x^2)/a;
```

```
>plot2d(expr,-2,2):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d(expr,-2,2): ...  
^
```

```
>plot2d(expr,r=1,thickness=2):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d(expr,r=1,thickness=2): ...  
^
```

```
>plot2d(&diff(expr,x),>add,style="--",color=red):
```

Variable red not found!

Error in:

```
plot2d(&diff(expr,x), >add, style="--", color=red): ...  
^
```

```
>plot2d(&diff(expr,x,2),a=-2,b=2,c=-2,d=1):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d(&diff(expr,x,2),a=-2,b=2,c=-2,d=1): ...  
^
```

```
>plot2d(&diff(expr,x),a=-2,b=2,>square):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d(&diff(expr,x),a=-2,b=2,&gt;square): ...  
^
```

```
>plot2d("x^2",0,1,steps=1,color=red,n=10):
```

```
Variable red not found!
```

```
Error in:
```

```
plot2d("x^2",0,1,steps=1,color=red,n=10): ...  
^
```

```
>plot2d("x^2",>add,steps=2,color=blue,n=10):
```

```
Variable blue not found!
```

```
Error in:
```

```
plot2d("x^2",&gt;add,steps=2,color=blue,n=10): ...  
^
```

## FUNGSI DALAM SATU PARAMETER

Fungsi plot yang paling penting untuk plot planar adalah plot2d(). Fungsi ini diimplementasikan dalam bahasa Euler di file “plot.e”, yang dimuat di awal program.

Berikut beberapa contoh penggunaan suatu fungsi. Seperti biasa di EMT, fungsi yang berfungsi untuk fungsi atau ekspresi lain, Anda bisa meneruskan parameter tambahan (selain x) yang bukan variabel global ke fungsi dengan parameter titik koma atau dengan kumpulan panggilan.

```
>function f(x,a) := x2/a+a*x2-x; // define a function
>a=0.3; plot2d("f",0,1;a): // plot with a=0.3
Function plot2d not found.
Try list ... to find functions!
Error in:
a=0.3; plot2d("f",0,1;a): // plot with a=0.3 ...
      ^

>plot2d("f",0,1;0.4): // plot with a=0.4
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("f",0,1;0.4): // plot with a=0.4 ...
      ^

>plot2d({{"f",0.2}},0,1): // plot with a=0.2
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d({{"f",0.2}},0,1): // plot with a=0.2 ...
      ^

>plot2d({{"f(x,b)",b=0.1}},0,1): // plot with 0.1
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d({{"f(x,b)",b=0.1}},0,1): // plot with 0.1 ...
```

```

>function f(x) := x^3-x; ...
> plot2d("f",r=1):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("f",r=1): ...

```

Berikut ini ringkasan fungsi yang diterima

- ekspresi atau ekspresi simbolik di x
- fungsi atau fungsi simbolik dengan nama "f"
- fungsi simbolik hanya dengan nama f

Fungsi plot2d() juga menerima fungsi simbolik. Untuk fungsi simbolik, namanya saja yang berfungsi.

```

>function f(x) &= diff(x^x,x)

```

$$x^x (\log(x) + 1)$$

```

>plot2d(f,0,2):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(f,0,2): ...

```

Tentu saja, untuk ekspresi atau ekspresi simbolik, nama variabel sudah cukup untuk memplotnya.

```

>expr &= sin(x)*exp(-x)

```

$$e^{-x} \sin(x)$$

```

>plot2d(expr,0,3pi):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(expr,0,3pi): ...

```

```

>function f(x) &= x^x;
>plot2d(f,r=1,cx=1,cy=1,color=blue,thickness=2);
Variable blue not found!
Error in:
plot2d(f,r=1,cx=1,cy=1,color=blue,thickness=2); ...
                                     ^

>plot2d(&diff(f(x),x),>add,color=red,style="-.-"):
Variable red not found!
Error in:
plot2d(&diff(f(x),x),>add,color=red,style="-.-")
                                     ^

```

Untuk gaya garis ada berbagai pilihan.

- gaya="...". Pilih dari "-", "-", "-.", ".", "-.", "-.-".
- Warna: Lihat di bawah untuk warna.
- ketebalan: Defaultnya adalah 1.

Warna dapat dipilih sebagai salah satu warna default, atau sebagai warna RGB.

- 0..15: indeks warna default.
- konstanta warna: putih, hitam, merah, hijau, biru, cyan, zaitun,
- abu-abu muda, abu-abu, abu-abu tua, oranye, hijau muda, pirus, biru
- muda, oranye muda, kuning
- rgb(merah,hijau,biru): parameternya real di [0,1].

```

>plot2d("exp(-x^2)",r=2,color=red,thickness=3,style="-"):
Variable red not found!
Error in:
plot2d("exp(-x^2)",r=2,color=red,thickness=3,style="-")
                                     ^

```

Berikut adalah tampilan warna EMT yang telah ditentukan sebelumnya.

```

>aspect(2); columnsplot(ones(1,16),lab=0:15,grid=0,color=0:15):
Function columnsplot not found.
Try list ... to find functions!

```

Error in:

```
... columnsplot(ones(1,16),lab=0:15,grid=0,color=0:15): ...  
^
```

Tapi Anda bisa menggunakan warna apa saja.

```
>columnsplot(ones(1,16),grid=0,color=rgb(0,0,linspace(0,1,15))):
```

Function rgb not found.

Try list ... to find functions!

Error in:

```
... ones(1,16),grid=0,color=rgb(0,0,linspace(0,1,15))): ...  
^
```



## MENG GAMBAR BEBERAPA KURVA PADA BIDANG KOORDINAT YANG SAMA

Memvisualisasikan data yang memiliki lebih dari satu fungsi ke dalam satu jenis jendela/gambar yang sama, dapat dilakukan dengan beberapa cara. Salah satu caranya adalah dengan menggunakan `>add`. `>add` ini digunakan untuk memanggil fungsi-fungsi tadi secara bersamaan. Kita telah menggunakan metode ini diatas, pada contoh contoh sebelumnya.

```
>aspect(); plot2d("cos(x)",r=2,grid=6); plot2d("x",style=".",>add):  
Built-in function aspect needs 1 argument (got 0)!  
Error in:  
aspect(); plot2d("cos(x)",r=2,grid=6); plot2d("x",sty
```

- Baris perintah diatas digunakan untuk membuat grafik dari fungsi
- `"cos(x)"`.
- `aspect()` digunakan untuk mengatur rasio dari grafik yang menentukan
- proporsi antara sumbu x dan sumbu y.
- `r=2`, artinya radius di sekitar pusat plotnya adalah 2
- `grid=6`, digunakan untuk mengatur jumlah garis grid atau jenis grid
- yang digunakan pada grafik.

Fungsi kedua yang akan kita plot adalah fungsi  $y=x$

- `style="."`, artinya kita ingin membuat grafik dengan menggunakan titik titik untuk menandai nilai-nilai pada grafik.

```
>aspect(2.5); plot2d("cos(x)",r=2,grid=6); plot2d("x",style="-",>add):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
aspect(2.5); plot2d("cos(x)",r=2,grid=6); plot2d("x",sty
```

```
>aspect(1.5); plot2d("sin(x)",0,2pi); plot2d("cos(x)",color=blue,style="--",>add):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
aspect(1.5); plot2d("sin(x)",0,2pi); plot2d("cos(x)",color
```

- Pada perintah ini kita menggunakan aspect(1.5). Fungsi yang ingin
- kita plot adalah fungsi sin(x) dengan nilai x=0 sampai x=2pi. Fungsi
- kedua yang akan kita tambahkan ke plot sin(x) adalah fungsi cos(x)
- dengan style garisnya berupa garis putus putus berwarna biru.

Salah satu kegunaan >add adalah untuk menambahkan titik pada kurva.

```
>plot2d("sin(x)",0,pi); plot2d(2,sin(2),>points,>add):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("sin(x)",0,pi); plot2d(2,sin(2),>points,>add)
```

- Pada perintah ini kita akan memplot grafik dari fungsi sin(x) dengan
- interval x dari 0 sampai pi. Lalu kita ingin menambahkan suatu titik
- pada grafik sinus x tadi, dengan cara menambahkan perintah lain yaitu
- plot2d(2, sin(2), >point, >add);. 2 dalam perintah tersebut artinya
- nilai x yang kita pilih adalah 2. Dengan menambahkan >point pada garis
- perintah, kita sudah dapat menambahkan satu titik pada grafik sinus
- tadi.

Pada contoh di bawah ini, kita akan memplot suatu fungsi dan menambahkan suatu titik dengan label nama ("cl" atau center left),

dan menyimpan hasilnya di notebook. Kita juga akan menambahkan label atau judul untuk plot fungsinya.

```
>plot2d(["cos(x)","x"],r=1.1,cx=0.5,cy=0.5,...
> color=[black,blue],style=["-","."],...
> grid=1);
Variable black not found!
Error in:
... "cos (x) ", "x" ], r=1.1, cx=0.5, cy=0.5,      color=[black,
...
>x0=solve("cos(x)-x",1);...
> plot2d(x0,x0,>points,>add,title="Intersection Demo");...
> label("cos(x) = x",x0,x0,pos="cl",offset=20):
Function solve not found.
Try list ... to find functions!
Error in:
x0=solve("cos (x) -x",1);      plot2d(x0,x0,>points,>add,
^
```

- Fungsi yang kita plot di atas adalah fungsi  $\cos(x)$  dan  $y=x$ , dengan
- rentang sumbu x dan sumbu y 1.1, artinya sumbu x dan y akan diperluas
- sedikit lebih dari -1.1 hingga 1.1, posisi pusat grafik pada koordinat
- adalah 0.5 relatif terhadap rentang grafik, kurva  $\cos(x)$  berwarna hitam dengan style garis solid dan kurva  $y=x$  berwarna biru dengan
- style titik-titik. Tampilan grid yang digunakan dalam plot ini adalah
- grid 1.
- $(x0=solve("cos(x)-x",1))$  merupakan perintah yang digunakan untuk
- menyelesaikan persamaan  $\cos(x)-x=0$ , dan hasilnya akan disimpan dalam

- variabel x0. Perintah plot yang kedua digunakan untuk menambahkan
- titik pada plot yang sudah dibuat sebelumnya, dengan nilai  $x=y=x0$ .
- Selain itu, plot kedua ini ditambahkan label nama “Intersection Demo”.
- (label(“cos(x)=x”,x0,x0,pos=“Cl”, offset=20) merupakan perintah
- untuk menambahkan label pada titik potong anatar grafik cos(x) dan x.
- Label nama ini diposisikan di center left, atau sebelah kiri pusat
- (titik) dan ada offset=20 dari titik potong, supaya titik tidak
- tertutup oleh label teks.

Dalam contoh berikut ini, kita akan memplot fungsi  $\text{sinc}(x)=\sin(x)/x$  dan ekspansi Taylor ke-8 dan ke-16. Untuk mencari ekspansi Taylor ini, kita akan menggunakan Maxima melalui ekspresi simbolik. Dalam perintah berikut, dilakukan pemanggilan plot2d() sebanyak 3 kali dan dilakukan dalam perintah multi baris. Perintah plot kedua dan ketiga memiliki set flag >add, yang membuat plot menggunakan nterval sebelumnya pada pemanggila plot pertama.

Kita menambahkan kotak label yang menjelaskan fungsi-sungsi tersebut.

```
>$taylor(sin(x)/x,x,0,4)
```

$$\frac{x^4}{120} - \frac{x^2}{6} + 1$$

```
>plot2d(“sinc(x)”,0,4pi,color=green,thickness=2); ...
```

```
> plot2d(&taylor(sin(x)/x,x,0,8),>add,color=blue,style=“-”); ...
```

```
> plot2d(&taylor(sin(x)/x,x,0,16),>add,color=red,style=“-.-”); ...
```

```
> labelbox([“sinc”,“T8”,“T16”],styles=[“-”,“-”,“-.-”], ...
```

```
> colors=[black,blue,red]):
```

```
Variable green not found!
```

```
Error in:
```

```
plot2d(“sinc(x)”,0,4pi,color=green,thickness=2);
```

```
plot2d(
```

- Pemanggilan plot pertama digunakan untuk menggambar fungsi  $\text{sinc}(x)$
- pada interval 0 sampai  $4\pi$ . Warna dari grafik ini adalah hijau, dengan
- ketebalan 2.
- Pemanggilan plot kedua digunakan untuk menggambar grafik dari
- polinomial Taylor, dari derajat 0 sampai dengan derajat 8. Warna dari
- grafik ini adalah biru, dan style garis putus putus.
- Pemanggilan plot ketiga digunakan untuk menggambar grafik dari
- polinomial Taylor, dari derajat 0 sampai dengan derajat 16. Warna dari
- grafik ini adalah biru, dan style garis dengan titik-titik putus.
- Perintah keempat digunakan untuk menambahkan kotak label untuk
- menjelaskan grafik-grafik yang digambar. Fungsi  $\text{sinc}(x)$  diberikan
- warna hitam dengan style garis solid, fungsi Taylor sampai derajat
- ke-8 diberikan warna biru dengan style garis putus-putus, dan fungsi
- Taylor sampai derajat ke-16 diberikann warna merah dengan style garis
- garis dengan titik-titik putus.

Pada contoh berikut, kami menghasilkan Polinomial Bernstein.

```
>plot2d("(1-x)^10",0,1): // plot first function
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("(1-x)^10",0,1): // plot first function ...
```

```

>for i=1 to 10; plot2d("bin(10,i)*xi*(1-x)(10-i)",>add); end;
Function plot2d not found.
Try list ... to find functions!
Error in:
... to 10; plot2d("bin(10,i)*xi*(1-x)(10-i)",>add);
^

>insimg;
Variable insimg not found!
Error in:
insimg; ...
^

```

- Perintah (plot2d("(1-x)<sup>10</sup>",0,1) digunakan untuk menggambarkan
- grafik dari fungsi (1-x)<sup>10</sup> pada rentang 0 hingga 1. Grafik ini
- menggambarkan bentuk polinomial dari fungsi tersebut.
- Perintah kedua digunakan untuk mengulang perintah dari i=1 hingga
- i=10. Dalam setiap iterasi loop, perintah ini menggambar grafik dari
- suku polinomial dalam ekspansi binomial dari (1-x)<sup>10</sup>. Fungsi yang
- diplot adalah bin(10,i).x<sup>i\*(1-x)</sup>(10-i) dimana bin(10,i) adalah
- koefisien binomial.
- Perintah insimg digunakan untuk menyisipkan atau menampilkan gambar
- dalam hasil output.

Metode kedua menggunakan sepasang matriks nilai x dan matriks nilai y dengan ukuran yang sama.

Kita membuat sebuah matriks nilai dengan satu Polinomial Bernstein di setiap baris. Untuk ini, kita cukup menggunakan vektor kolom i. Baca kembali pengantar tentang bahasa matriks untuk mempelajari lebih lanjut.

```

>x=linspace(0,1,500);
>n=10; k=(0:n)'; // n is row vector, k is column vector

```

```

>y=bin(n,k)*xk*(1-x)(n-k); // y is a matrix then
>plot2d(x,y):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(x,y): ...
^

```

- Perintah linspace digunakan untuk membuat vektor x yang berisi 500
- titik yang terdistribusi secara merata (linier) dari 0 hingga 1.
- Perintah kedua digunakan untuk membuat vektor.
- Perintah ketiga digunakan untuk menghitung nilai-nilai untuk y
- berdasarkan ekspansi binomial. Hasil dari operasi pada baris perintah
- ini adalah sebuah matriks.
- plot2d(x,y) digunakan untuk menggambar grafik 2d dari matriks y
- terhadap vektor x.

Perhatikan bahwa parameter warna dapat berupa vektor. Kemudian setiap warna digunakan untuk setiap baris matriks.

```

>x=linspace(0,1,200); y=x^(1:10)'; plot2d(x,y,color=1:10):
Function plot2d not found.
Try list ... to find functions!
Error in:
... ace(0,1,200); y=x^(1:10)'; plot2d(x,y,color=1:10)

```

Cara lain adalah dengan menggunakan vektor ekspresi (string). Lalu kalian dapat menggunakan larik warna, larik gaya, dan larik ketebalan dengan panjang yang sama.

```

>plot2d(["sin(x)","cos(x)"],0,2pi,color=2:3):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(["sin(x)", "cos(x)"], 0, 2pi, color=2:3): ...

```

Perintah diatas digunakan untuk memplot fungsi  $\sin(x)$  dan  $\cos(x)$  dalam rentang 0 hingga  $2\pi$ . Pada plot in, kedua fungsi diberikan warna yang berbeda.

```
>plot2d(["sin(x)","cos(x)"],0,2pi): // plot vector of expressions
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(["sin(x)","cos(x)"],0,2pi): // plot vector of expressions
```

Kita bisa mendapatkan vektor seperti itu dari Maxima menggunakan makelist() dan mxm2str().

```
>v &= makelist(binomial(10,i)*xi*(1-x)(10-i),i,0,10) // make list
```

$$\left[ \binom{10}{0} (1-x)^{10}, \binom{10}{1} (1-x)^9 x, \binom{10}{2} (1-x)^8 x^2, \binom{10}{3} (1-x)^7 x^3, \binom{10}{4} (1-x)^6 x^4, \binom{10}{5} (1-x)^5 x^5, \binom{10}{6} (1-x)^4 x^6, \binom{10}{7} (1-x)^3 x^7, \binom{10}{8} (1-x)^2 x^8, \binom{10}{9} (1-x) x^9, x^{10} \right]$$

- fungsi makelist() digunakan untuk membuat list dari elemen-elemen
- yang dihasilkan oleh ekspresi di dalamnya berdasarkan parameter yang
- diberikan.
- $(\text{binomial}(10,i))$  merupakan fungsi yang digunakan untuk menghitung
- koefisien binomial, yaitu angka yang muncul dalam ekspansi dari
- $(a+b)^{10}$ .
- $(i,0,10)$  merupakan rentang iterasi untuk  $i$  dalam fungsi makelist.
- Artinya  $i$  akan berubah dari 0 hingga 10 dan untuk setiap nilai  $i$
- dalam rentang ini, ekspresi  $\text{binomial}(10,i)x^i(1-x)^{(10-i)}$  akan
- dihitung dan dimasukkan ke dalam list.



```
>mxm2str(v) // get a vector of strings from the symbolic vector
Function mxm2str not found.
Try list ... to find functions!
Error in:
mxm2str(v) // get a vector of strings from the symbol
^
```

- mxmstr() merupakan fungsi yang digunakan untuk mengkonversi atau
- mengubah format data dari bentuk simbolik menjadi bentuk string.

```
>plot2d(mxm2str(v),0,1): // plot functions
Function mxm2str not found.
Try list ... to find functions!
Error in:
plot2d(mxm2str(v),0,1): // plot functions ...
^
```

- Perintah diatas digunakan untuk memvisualisasikan fungsi yang sudah
- dikonversi sebelumnya, yaitu fungsi mxm2str(v), dari rentang  $x=0$
- sampai  $x=1$ .

Alternatif lain adalah dengan menggunakan bahasa matriks di Euler.

Jika sebuah ekspresi menghasilkan sebuah matriks fungsi, dengan satu fungsi di setiap baris, semua fungsi ini akan diplot ke dalam satu plot.

Untuk ini, gunakan vektor parameter dalam bentuk vektor kolom. Jika sebuah larik warna ditambahkan, maka akan digunakan untuk setiap baris plot.

```
>n=(1:10)'; plot2d("x^n",0,1,color=1:10):
Function plot2d not found.
Try list ... to find functions!
Error in:
n=(1:10)'; plot2d("x^n",0,1,color=1:10): ...
^
```

Ekspresi dan fungsi satu baris dapat melihat variabel global.

Jika kalian tidak dapat menggunakan variabel global, kalian perlu menggunakan fungsi dengan parameter ekstra, dan memberikan parameter ini sebagai parameter semicolon atau titik koma.

Hati-hati dalam meletakkan semua parameter yang diberikan di akhir perintah plot2d. Pada contoh di bawah ini, kita memasukkan nilai  $a=5$  ke dalam fungsi  $f$ , yang kita plot dari -10 hingga 10.

```
>function f(x,a) := 1/a*exp(-x^2/a); ...
> plot2d("f",-10,10;5,thickness=2,title="a=5"):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("f",-10,10;5,thickness=2,title="a=5"): ...
```

- Plot diatas merupakan plot dari fungsi  $f(x,a) := 1/a \cdot \exp(-x^2/a)$ ,  
x
- merupakan variabel independen dan a merupakan parameter yang
- mempengaruhi bentuk fungsi. Interval sumbu x dari  $x = -10$  hingga  $x=10$ ,
- ketebalan garis kurvana 2, dengan judul grafik "a=5".

Atau gunakan koleksi dengan nama fungsi dan semua parameter tambahan. List atau daftar khusus ini disebut koleksi panggilan, dan itu merupakan cara yang lebih banyak digunakan untuk mengoper argumen ke fungsi yang dengan sendirinya dioper sebagai argumen ke fungsi lain.

Pada contoh berikut, kita menggunakan perulangan untuk memplot beberapa fungsi.

```
>plot2d({{"f",1}},-10,10); ...
> for a=2:10; plot2d({{"f",a}},>add); end:
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d({{"f",1}},-10,10); for a=2:10; plot2d({{"f",a}},>add); end
```

- Plot diatas merupakan plot dari fungsi  $f$  dengan parameter  $a=1$ , dan

- dari interval  $x=-10$  sampai  $x=10$ .
- Perintah kedua merupakan perulangan untuk menggambar grafik fungsi  $f$
- dengan parameter  $a=2:10$  (2 sampai 10). Grafik baru dengan nilai  $a=2:10$
- ini kemudian ditambahkan ke grafik pertama saat nilai  $a=1$ .

Kita dapat mendapatkan hasil yang sama dengan grafik diatas menggunakan cara berikut, yaitu menggunakan bahasa matriks EMT. Masing-masing matriks  $f(x,a)$  adalah satu fungsi. Selain itu, kita dapat mengatur masing-masing baris dari matriks menggunakan warna yang berbeda. Klik dua kali pada fungsi `getspectral()` untuk penjelasan lebih lanjut.

```
>x=-10:0.01:10;a=(1:10)'; plot2d(x,f(x,a),color=getspectral(a/10));
Function getspectral not found.
Try list ... to find functions!
Error in:
... =(1:10)'; plot2d(x,f(x,a),color=getspectral(a/10))
```

- Perintah  $x=-10:0.01:10$  digunakan untuk membuat vektor  $x$  yang berisi
- nilai dari -10 hingga 10 dengan interval 0.01.
- Perintah  $a=(1:10)'$  digunakan untuk mendefinisikan vektor  $a$  yang
- berisi nilai dari 10 hingga 10.
- Perintah `plot2d(x,f(x,a),color=getspectral(a/10))` digunakan untuk
- menggambar grafik dari fungsi  $f$  dengan  $x$  sebagai variabel independen
- dan  $a$  sebagai parameter. `getspectral()` digunakan untuk menetapkan
- warna yang berbeda pada setiap kurva.

### Soal Latihan Tambahan

1. Sketsakan grafik fungsi berikut di interval 1:10

```

>function g(x) := sqrt((x+3)+a^(a-1)); ...
> for a=1:5; plot2d("g",1,10,title="Grafik g(x)"); end:
Function plot2d not found.
Try list ... to find functions!
Error in:
for a=1:5; plot2d("g",1,10,title="Grafik g(x)"); end: ...

```

2. Carilah grafik dari fungsi berikut pada interval  $[-\pi, 2\pi]$

```

>function y(t) := sin(t-(t/4)); ...
> plot2d("y",-pi,2pi,color=blue,title="Grafik y(t)"):
Variable blue not found!
Error in:
plot2d("y",-pi,2pi,color=blue,title="Grafik y(t)"): ...

```

## Label Teks

Dekorasi sederhana pun bisa

- judul dengan title= "..."
- label x dan y dengan xl="...", yl="..."
- label teks lain dengan label("...",x,y)

Perintah label akan memplot ke plot saat ini pada koordinat plot (x,y). Hal ini memerlukan argumen posisional.

```

>plot2d("x^3-x",-1,2,title="y=x^3-x",yl="y",xl="x"):
Function plot2d not found.
Try list ... to find functions!
Error in:
... lot2d("x^3-x",-1,2,title="y=x^3-x",yl="y",xl="x"): ...

```

```

>expr := "log(x)/x"; ...
> plot2d(expr,0.5,5,title="y="+expr,xl="x",yl="y"); ...
> label("(1,0)",1,0); label("Max",E,expr(E),pos="lc"):
Function plot2d not found.
Try list ... to find functions!
Error in:
... plot2d(expr,0.5,5,title="y="+expr,xl="x",yl="y");

```

Ada juga fungsi `labelbox()`, yang dapat menampilkan fungsi dan teks. Dibutuhkan vektor string dan warna, satu item untuk setiap fungsi.

```
>function f(x) &= x2*exp(-x2); ...
> plot2d(&f(x),a=-3,b=3,c=-1,d=1); ...
> plot2d(&diff(f(x),x),>add,color=blue,style="-"); ...
> labelbox(["function","derivative"],styles=["-","-"], ...
> colors=[black,blue],w=0.4):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(&f(x), a=-3, b=3, c=-1, d=1); plot2d(&diff(x), a=-3, b=3, c=-1, d=1);
```

Kotak ini berlabuh di kanan atas secara default, tetapi `>kiri` berlabuh di kiri atas. Anda dapat memindahkannya ke tempat mana pun yang Anda suka. Posisi jangkar berada di pojok kanan atas kotak, dan angkanya merupakan pecahan dari ukuran jendela grafis. Lebarnya otomatis.

Untuk plot titik, kotak label juga berfungsi. Tambahkan parameter `>points`, atau vektor bendera, satu untuk setiap label.

Pada contoh berikut, hanya ada satu fungsi. Jadi kita bisa menggunakan string sebagai pengganti vektor string. Kami mengatur warna teks menjadi hitam untuk contoh ini.

```
>n=10; plot2d(0:n,bin(n,0:n),>addpoints); ...
> labelbox("Binomials",styles="[]",>points,x=0.1,y=0.1, ...
> tcolor=black,>left):
Function plot2d not found.
Try list ... to find functions!
Error in:
n=10; plot2d(0:n,bin(n,0:n),>addpoints); labelbox("Binomials",styles="[]",x=0.1,y=0.1,tcolor=black,align="left");
```

Gaya plot ini juga tersedia di `statplot()`. Seperti di `plot2d()` warna dapat diatur untuk setiap baris plot. Masih banyak lagi plot khusus untuk keperluan statistik (lihat tutorial tentang statistik).

```
>statplot(1:10,random(3,10),color=[red,blue, green]):
```

Variable red not found!

Error in:

```
statplot(1:10,random(3,10),color=[red,blue, green]): ...  
^
```

Fitur serupa adalah fungsi `textbox()`.

Lebar nya secara default adalah lebar maksimal baris teks. Tapi itu bisa diatur oleh pengguna juga.

```
>function f(x) &= exp(-x)*sin(2*pi*x); ...  
> plot2d("f(x)",0,2pi); ...  
> textbox(latex("\text{Example of a damped oscillation}\ f(x)=e^{-x}\sin(2\pi x)"),w=0.85):
```

Function `plot2d` not found.

Try list ... to find functions!

Error in:

```
plot2d("f(x)",0,2pi); textbox(latex("\text{Example of a da  
^
```

Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

```
>plot2d("x^3-x",title="x → x3 - x"):
```

Function `plot2d` not found.

Try list ... to find functions!

Error in:

```
plot2d("x^3-x",title="x & rarr; x&sup3; - x"): ...  
^
```

Label pada sumbu x dan y bisa vertikal, begitu juga dengan sumbunya.

```
>plot2d("sinc(x)",0,2pi,xl="x",yl="x → sinc(x)",>vertical):
```

Function `plot2d` not found.

Try list ... to find functions!

Error in:

```
... )",0,2pi,xl="x",yl="x & rarr; sinc(x)",>vertical  
^
```

## LaTeX

Anda juga dapat memplot rumus LaTeX jika Anda telah menginstal sistem LaTeX. Saya merekomendasikan MiKTeX. Jalur ke biner “latex” dan “dvi2ps” harus berada di jalur sistem, atau Anda harus mengatur LaTeX di menu opsi.

Perhatikan, penguraian LaTeX lambat. Jika Anda ingin menggunakan LaTeX dalam plot animasi, Anda harus memanggil `latex()` sebelum loop satu kali dan menggunakan hasilnya (gambar dalam matriks RGB).

Pada plot berikut, kami menggunakan LaTeX untuk label  $x$  dan  $y$ , label, kotak label, dan judul plot.

```
>plot2d("exp(-x)*sin(x)/x",a=0,b=2pi,c=0,d=1,grid=6,color=blue,
...
> title=latex("\text{Function}
Phi}"),...
> xl=latex("\phi"),yl=latex("\Phi(\phi)"));...
> textbox(...
> latex("\Phi(\phi) = e^{-\phi} \frac{\sin(\phi)}{\phi}"),x=0.8,y=0.5);
...
> label(latex("\Phi",color=blue),1,0.4):
Variable blue not found!
Error in:
... -x)*sin(x)/x",a=0,b=2pi,c=0,d=1,grid=6,color=blue,
```

Seringkali, kita menginginkan spasi dan label teks yang tidak konformal pada sumbu  $x$ . Kita bisa menggunakan `xaxis()` dan `yaxis()` seperti yang akan kita tunjukkan nanti.

Cara termudah adalah membuat plot kosong dengan bingkai menggunakan `grid=4`, lalu menambahkan grid dengan `ygrid()` dan `xgrid()`. Pada contoh berikut, kami menggunakan tiga string LaTeX untuk label pada sumbu  $x$  dengan `xtick()`.

```
>plot2d("sinc(x)",0,2pi,grid=4,<ticks);...
> ygrid(-2:0.5:2,grid=6);...
> xgrid([0:2]*pi,<ticks,grid=6);...
> xtick([0,pi,2pi],["0","\pi","2\pi"],>latex):
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("sinc(x)",0,2pi,grid=4,&lt;ticks); ygrid(-2:0.5:2,g
```

Tentu saja fungsinya juga bisa digunakan.

```
>function map f(x)...
if x>0 then return x^4
else return x^2
endif
endfunction
```

Parameter “map” membantu menggunakan fungsi untuk vektor.  
Untuk

plot, itu tidak perlu. Tapi untuk menunjukkan vektorisasi itu berguna, kita menambahkan beberapa poin penting ke plot di  $x=-1$ ,  $x=0$  dan  $x=1$ .

Pada plot berikut, kami juga memasukkan beberapa kode LaTeX. Kami menggunakannya untuk

dua label dan kotak teks. Tentu saja, Anda hanya bisa menggunakannya

LaTeX jika Anda telah instal LaTeX dengan benar.

```
>plot2d("f",-1,1,xl="x",yl="f(x)",grid=6);...
> plot2d([-1,0,1],f([-1,0,1]),>points,>add); ...
> label(latex("x^3"),0.72,f(0.72)); ...
> label(latex("x^2"),-0.52,f(-0.52),pos="l"); ...
> textbox( ...
> latex("f(x)=\begin{cases} x^3 & x>0 \\ x^2 & x \le 0 \end{cases}"),
...
> x=0.7,y=0.2):
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("f",-1,1,xl="x",yl="f(x)",grid=6); plot2d([-1,0,1]
```



## Interaksi Pengguna

Saat memplot suatu fungsi atau ekspresi, parameter `>pengguna` memungkinkan pengguna untuk memperbesar dan menggeser plot dengan tombol kursor atau mouse. Pengguna bisa

- perbesar dengan `+` atau `-`
- pindahkan plot dengan tombol kursor
- pilih jendela plot dengan mouse
- atur ulang tampilan dengan spasi
- keluar dengan kembali

Tombol spasi akan mengatur ulang plot ke jendela plot aslinya.

Saat memplot data, flag `>user` hanya akan menunggu penekanan tombol.

```
>plot2d({{"x^3-a*x",a=1}},>user,title="Press any key!"):
Function plot2d not found.
Try list ... to find functions!
Error in:
... d({{"x^3-a*x",a=1}},&gt;user,title="Press any key

>plot2d("exp(x)*sin(x)",user=true,...
> title="+/- or cursor keys (return to exit)"):
Function plot2d not found.
Try list ... to find functions!
Error in:
... e,   title="+/- or cursor keys (return to exit) ")
```

Berikut ini menunjukkan cara interaksi pengguna tingkat lanjut (lihat tutorial tentang pemrograman untuk detailnya).

Fungsi bawaan `mousedrag()` menunggu aktivitas mouse atau keyboard. Ini melaporkan mouse ke bawah, mouse digerakkan atau mouse ke atas, dan penekanan tombol. Fungsi `dragpoints()` memanfaatkan ini, dan memungkinkan pengguna menyeret titik mana pun dalam plot.

Kita membutuhkan fungsi plot terlebih dahulu. Misalnya, kita melakukan interpolasi pada 5 titik dengan polinomial. Fungsi tersebut

harus diplot ke dalam area plot yang tetap.

```
>function plotf(xp,yp,select) ...
d=interp(xp,yp);
plot2d("interpval(xp,d,x)";d,xp,r=2);
plot2d(xp,yp,>points,>add);
if select>0 then
  plot2d(xp[select],yp[select],color=red,>points,>add);
endif;
title("Drag one point, or press space or return!");
endfunction
```

Perhatikan parameter titik koma di plot2d (d dan xp), yang diteruskan ke evaluasi fungsi interp(). Tanpa ini, kita harus menulis fungsi plotinterp() terlebih dahulu, mengakses nilainya secara global.

Sekarang kita menghasilkan beberapa nilai acak, dan membiarkan pengguna menyeret titiknya.

```
>t=-1:0.5:1; dragpoints("plotf",t,random(size(t))-0.5):
Function dragpoints not found.
Try list ... to find functions!
Error in:
... :0.5:1; dragpoints("plotf",t,random(size(t))-0.5): ...
```

Ada juga fungsi yang memplot fungsi lain bergantung pada vektor parameter, dan memungkinkan pengguna menyesuaikan parameter ini.

Pertama kita membutuhkan fungsi plot.

```
>function plotf([a,b]) := plot2d("exp(a*x)*cos(2pi*b*x)",0,2pi;a,b):
```

Kemudian kita memerlukan nama untuk parameter, nilai awal dan matriks rentang nx2, opsional garis judul.

Ada penggeser interaktif, yang dapat menetapkan nilai oleh pengguna. Fungsi dragvalues() menyediakan ini.

```
>dragvalues("plotf",["a","b"],[-1,2],[[-2,2];[1,10]], ...
> heading="Drag these values:",hcolor=black):
Variable black not found!
Error in:
... 10]], heading="Drag these values:",hcolor=black): ..
```

Dimungkinkan untuk membatasi nilai yang diseret menjadi bilangan bulat. Sebagai contoh, kita menulis fungsi plot, yang memplot polinomial Taylor berderajat n ke fungsi kosinus.

```
>function plotf(n) ...
plot2d("cos (x) ", 0, 2pi, >square, grid=6);
plot2d("&taylor(cos (x) , x, 0, @n) ", color=blue, >add);
textbox("Taylor polynomial of degree "+n, 0.1, 0.02, sty
endfunction
```

```
>plotf(1):
Function plot2d not found.
Try list ... to find functions!
Try "trace errors" to inspect local variables after e
plotf:
```

```
plot2d("cos (x) ", 0, 2pi, &gt;square, grid=6);
```

Sekarang kita izinkan derajat n bervariasi dari 0 hingga 20 dalam 20 perhentian. Hasil dragvalues() digunakan untuk memplot sketsa dengan n ini, dan untuk memasukkan plot ke dalam buku catatan.

```
>nd=dragvalues("plotf","degree",2,[0,20],20,y=0.8, ...
> heading="Drag the value:"); ...
> plotf(nd):
Function dragvalues not found.
Try list ... to find functions!
Error in:
... , 2, [0, 20], 20, y=0.8, heading="Drag the value:")
```

Berikut ini adalah demonstrasi sederhana dari fungsinya. Pengguna dapat menggambar jendela plot, meninggalkan jejak titik.

```
>function dragtest ...
plot2d(none, r=1, title="Drag with the mouse, or pres
start=0;
repeat
{flag, m, time}=mousedrag();
if flag==0 then return; endif;
if flag==2 then
hold on; mark(m[1], m[2]); hold off;
```

```

    endif;
end
endfunction

>dragtest // lihat hasilnya dan cobalah lakukan!
Function plot2d not found.
Try list ... to find functions!
Try "trace errors" to inspect local variables after errors
dragtest:
    plot2d(none,r=1,title="Drag with the mouse, or press a

```

## Gaya Plot 2D

Secara default, EMT menghitung penanda kecil sumbu otomatis dan menambahkan label ke setiap penanda. Ini dapat diubah dengan parameter tampilan. Gaya default sumbu dan label dapat diubah. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk menyetel ulang ke gaya default, gunakan reset().

```

>aspect();
Built-in function aspect needs 1 argument (got 0)!
Error in:
aspect(); ...
    ^

>figure(3,4); ...
> figure(1); plot2d("x^3-x",grid=0); ... // tidak ada tampilan, bingkai
dan sumbu
Function figure not found.
Try list ... to find functions!
Error in:
figure(3,4);  figure(1); plot2d("x^3-x",grid=0); ... // ti
    ^

> figure(2); plot2d("x^3-x",grid=1); ... // terdapat sumbu x-y
Function figure not found.
Try list ... to find functions!
Error in:
    figure(2); plot2d("x^3-x",grid=1); ... // terdapat sumbu
        ^

> figure(3); plot2d("x^3-x",grid=2); ... // terdapat penanda

```

kecil otomatis

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
  figure(3); plot2d("x^3-x",grid=2); ... // terdapat penanda  
    ^
```

```
    > figure(4); plot2d("x^3-x",grid=3); ... // sumbu x-y dengan  
label di dalamnya
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
  figure(4); plot2d("x^3-x",grid=3); ... // sumbu x-y dengan  
    ^
```

```
    > figure(5); plot2d("x^3-x",grid=4); ... // tidak ada penanda  
hanya label
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
  figure(5); plot2d("x^3-x",grid=4); ... // tidak ada penanda  
    ^
```

```
    > figure(6); plot2d("x^3-x",grid=5); ... // default tapi tidak ada  
margin
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
  figure(6); plot2d("x^3-x",grid=5); ... // default tapi tidak ada  
    ^
```

```
    > figure(7); plot2d("x^3-x",grid=6); ... // hanya sumbu dan  
penanda kecil
```

```
Function figure not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
  figure(7); plot2d("x^3-x",grid=6); ... // hanya sumbu dan penanda  
    ^
```

```
    > figure(8); plot2d("x^3-x",grid=7); ... // hanya sumbu dan  
penanda kecil pada sumbu
```

Function figure not found.

Try list ... to find functions!

Error in:

```
figure(8); plot2d("x^3-x",grid=7); ... // hanya sumbu dan  
^
```

> figure(9); plot2d("x^3-x",grid=8); ... // hanya sumbu dan  
penanda kecil terperinci pada sumbu tertentu

Function figure not found.

Try list ... to find functions!

Error in:

```
figure(9); plot2d("x^3-x",grid=8); ... // hanya sumbu dan  
^
```

> figure(10); plot2d("x^3-x",grid=9); ... // default dengan  
penanda-penanda kecil di dalamnya

Function figure not found.

Try list ... to find functions!

Error in:

```
figure(10); plot2d("x^3-x",grid=9); ... // default dengan  
^
```

> figure(11); plot2d("x^3-x",grid=10); ...// tidak ada penanda  
kecil, hanya sumbu

Function figure not found.

Try list ... to find functions!

Error in:

```
figure(11); plot2d("x^3-x",grid=10); ...// tidak ada pena  
^
```

> figure(0):

Function figure not found.

Try list ... to find functions!

Error in:

```
figure(0): ...  
^
```

Parameter <frame mematikan frame, dan framecolor=blue  
mengatur frame menjadi warna biru.

Jika Anda menginginkan tanda penanda Anda sendiri, Anda  
dapat menggunakan style=0, dan menambahkan semuanya nanti.

```

>aspect(1.5);
>plot2d("x^3-x",grid=0); // plot
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("x^3-x",grid=0); // plot ...
      ^

```

```

>frame; xgrid([-1,0,1]); ygrid(0): // add frame and grid
Function xgrid not found.
Try list ... to find functions!
Error in:
frame; xgrid([-1,0,1]); ygrid(0): // add frame and gr
      ^

```

Untuk judul plot dan label sumbu, lihat contoh berikut.

```

>plot2d("exp(x)",-1,1);
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("exp(x)",-1,1); ...
      ^

```

```

>textcolor(black); // set the text color to black
Variable or function black not found.
Error in:
textcolor(black); // set the text color to black ...
      ^

```

```

>title(latex("y=e^x")); // title above the plot
Function latex not found.
Try list ... to find functions!
Error in:
title(latex("y=e^x")); // title above the plot ...
      ^

```

```

>xlabel(latex("x")); // "x" for x-axis
Function latex not found.
Try list ... to find functions!
Error in:

```

```
xlabel(latex("x")); // "x" for x-axis ...
^
```

```
>ylabel(latex("y"),>vertical); // vertical "y" for y-axis
```

Function latex not found.

Try list ... to find functions!

Error in:

```
ylabel(latex("y"),>vertical); // vertical "y" for y-axis
^
```

```
>label(latex("(0,1)"),0,1,color=blue): // label a point
```

Function latex not found.

Try list ... to find functions!

Error in:

```
label(latex("(0,1)"),0,1,color=blue): // label a point ...
^
```

Sumbu dapat digambar secara terpisah dengan xaxis() dan yaxis().

```
>plot2d("x^3-x",<grid,<frame);
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("x^3-x",&lt;grid,&lt;frame); ...
^
```

```
>xaxis(0,xx=-2:1,style="->"); yaxis(0,yy=-5:5,style="->");
```

Function xaxis not found.

Try list ... to find functions!

Error in:

```
xaxis(0,xx=-2:1,style="->"); yaxis(0,yy=-5:5,style="->")
^
```

Teks pada plot dapat diatur dengan label(). Dalam contoh berikut, "lc" berarti bagian tengah bawah. Ini menetapkan posisi label relatif terhadap koordinat plot.

```
>function f(x) &= x^3-x
```

$$\begin{array}{c} 3 \\ x^3 - x \end{array}$$

```
>plot2d(f,-1,1,>square);
```



```
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(f,-1,1,&gt;square); ...
      ^
```

```
>x0=fmin(f,0,1); // compute point of minimum
```

```
Function fmin not found.
Try list ... to find functions!
Error in:
x0=fmin(f,0,1); // compute point of minimum ...
      ^
```

```
>label("Rel. Min.",x0,f(x0),pos="lc"): // add a label there
```

```
Variable or function x0 not found.
Error in:
label("Rel. Min.",x0,f(x0),pos="lc"): // add a label there
      ^
```

```
Ada juga kotak teks.
```

```
>plot2d(f,-1,1,-2,2); // function
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(f,-1,1,-2,2); // function ...
      ^
```

```
>plot2d(&diff(f(x),x),>add,style="--",color=red); // derivative
```

```
Variable red not found!
Error in:
plot2d(&diff(f(x),x),&gt;add,style="--",color=red);
      ^
```

```
>labelbox(["f","f'"],["-","--"],[black,red]): // label box
```

```
Variable black not found!
Error in:
labelbox(["f","f'"],["-","--"],[black,red]): // label box
      ^
```

```
>plot2d(["exp(x)","1+x"],color=[black,blue],style=["-","-."]):
```

```
Variable black not found!
```

Error in:

```
plot2d(["exp(x)", "1+x"], color=[black, blue], style=["-", "-."],  
      ^  
      >gridstyle("-", color=gray, textcolor=gray, framecolor=gray);  
...  
> plot2d("x^3-x", grid=1); ...  
> settitle("y=x^3-x", color=black); ...  
> label("x", 2, 0, pos="bc", color=gray); ...  
> label("y", 0, 6, pos="cl", color=gray); ...  
> reset():  
Variable gray not found!  
Error in:  
gridstyle("-", &gt; ", color=gray, textcolor=gray, framecolor=gray",  
          ^
```

Untuk kontrol lebih lanjut, sumbu x dan sumbu y dapat dilakukan secara manual.

Perintah `fullwindow()` memperluas jendela plot karena kita tidak lagi memerlukan tempat untuk label di luar jendela plot. Gunakan `shrinkwindow()` atau `reset()` untuk menyetel ulang ke default.

```
> fullwindow; ...  
> gridstyle(color=darkgray, textcolor=darkgray); ...  
> plot2d(["2^x", "1", "2(-x)"], a=-2, b=2, c=0, d=4, <grid, color=4:6, <frame);  
...  
> xaxis(0, -2:1, style="->"); xaxis(0, 2, "x", <axis); ...  
> yaxis(0, 4, "y", style="->"); ...  
> yaxis(-2, 1:4, >left); ...  
> yaxis(2, 2^(-2:2), style=".", <left); ...  
> labelbox(["2^x", "1", "2(-x)"], colors=4:6, x=0.8, y=0.2); ...  
> reset:  
Variable fullwindow not found!  
Error in:  
fullwindow; gridstyle(color=darkgray, textcolor=darkgray);  
          ^
```

Berikut adalah contoh lain, di mana string Unicode digunakan dan sumbunya berada di luar area plot.

```
> aspect(1.5);
```

```

>plot2d(["sin(x)","cos(x)"],0,2pi,color=[red,green],<grid,<frame);
...
> xaxis(-1.1,(0:2)*pi,xt=["0",u""u"2"],style="-",>ticks,>zero); ...
> xgrid((0:0.5:2)*pi,<ticks); ...
> yaxis(-0.1*pi,-1:0.2:1,style="-",>zero,>grid); ...
> labelbox(["sin","cos"],colors=[red,green],x=0.5,y=0.2,>left); ...
> xlabel(u""); ylabel(u"f()"):
Variable red not found!
Error in:
plot2d(["sin(x) ", "cos(x) "], 0, 2pi, color=[red, green], &l

```



## MEMPLOT DATA 2D

Jika  $x$  dan  $y$  adalah vektor data, maka data dalam vektor  $x$  dan  $y$  ini akan digunakan sebagai koordinat  $x$  dan  $y$  dari sebuah kurva. Dalam hal ini,  $a$ ,  $b$ ,  $c$ , dan  $d$ , atau radius  $r$  dapat ditentukan, jika nilai-nilai tersebut tidak ditentukan, plot window akan menyesuaikan secara otomatis dengan data. Sebagai alternatif, perintah `>square` dapat digunakan untuk mempertahankan rasio aspek persegi.

Memplot sebuah ekspresi hanyalah singkatan untuk plot data. Untuk data plot, kalian membutuhkan satu atau beberapa baris nilai  $x$ , dan satu atau beberapa baris nilai  $y$ . Dari rentang dan nilai  $x$ , fungsi `plot2d` akan menghitung data untuk diplot secara default dengan evaluasi adaptif dari fungsi tersebut. Untuk menambahkan titik pada gambar grafik, gunakan perintah `">points"`, untuk garis dan titik gunakan perintah `">addpoints"`.

Namun, kalian juga dapat memasukkan data secara langsung.

- Gunakan vektor baris untuk  $x$  dan  $y$  untuk satu fungsi
- Matriks untuk  $x$  dan  $y$  diplot baris demi baris

Berikut adalah contoh dengan satu baris untuk  $x$  dan  $y$ .

```
>x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y): ...
```

- `x=-10:0.1:10`, merupakan perintah untuk membuat sebuah vektor dengan
- nama  $x$ , yang berisi deretan nilai dari -10 hingga 10 dengan beda
- setiap nilainya adalah 0.1.
- `y=exp(-x^2)*x`, merupakan perintah untuk membuat vektor dengan nama
- $y$ , yang berisi deretan nilai berdasarkan nilai  $x$ .

- `plot2d(x,y)`, merupakan perintah untuk membuat grafik 2d dari data x

- dan y.

Data juga bisa diplot sebagai titik. Gunakan perintah `points=true` untuk membuatnya. Plot ini bekerja seperti poligon, namun hanya menggambar sudut-sudutnya saja.

- `style="..."`: Select from `"[]"`, `"<>"`, `"o"`, `"."`, `".."`, `"+"`, `"*"`,
- `"[]#"`, `"<>#"`, `"o#"`, `"..#"`, `"#"`, `"|"`.

Untuk memplot kumpulan titik, gunakan perintah `>points`. Jika warnanya merupakan vektor warna, setiap titik akan mendapatkan warna yang berbeda. Untuk matriks koordinat dan vektor kolom, warna tersebut diaplikasikan pada baris setiap matriks. Parameter `>addpoints` menambahkan titik-titik ke segmen garis untuk plot data.

```
>xdata=[1,1.5,2.5,3,4]; ydata=[3,3.1,2.8,2.9,2.7]; // data
>plot2d(xdata,ydata,a=0.5,b=4.5,c=2.5,d=3.5,style="."): //
```

lines

Function `plot2d` not found.

Try list ... to find functions!

Error in:

```
... 2d(xdata,ydata,a=0.5,b=4.5,c=2.5,d=3.5,style="."): //
^
```

Perintah diatas digunakan untuk membuat plot 2d dari `xdata` dan `ydata`, dengan rentang sumbu x dari 0.5 - 4.5 dan rentang sumbu y dari 2.5 - 3.5. Kumpulan titik-titik data tersebut akan ditamapilkan dengan style titik.

```
>plot2d(xdata,ydata,>points,>add,style="o"): // add points
```

Function `plot2d` not found.

Try list ... to find functions!

Error in:

```
plot2d(xdata,ydata,&gt;points,&gt;add,style="o"): // add p
^
```

```
>p=polyfit(xdata,ydata,1); // get regression line
```

Function `polyfit` not found.

Try list ... to find functions!

Error in:

```
p=polyfit(xdata,ydata,1); // get regression line ...
^
```

Perintah diatas digunakan untuk menghitung koefisien polinomial orde 1, dimana hasilnya adalah nilai p yang berisi dua koefisien dari garis regresi. Dua koefisien ini merupakan kemiringan dan intercept dari garis regresi. Garis regresi adalah garis lurus yang menggambarkan hubungan antara dua variabel, yaitu variabel dependen (y) dan variabel independen (x). Garis regresi digunakan untuk menggambarkan perilaku sekumpulan data.

```
>plot2d("polyval(p,x)",>add,color=red): // add plot of line
Variable red not found!
Error in:
plot2d("polyval(p,x)", &gt;add,color=red): // add plot
^
```

Perintah ini digunakan untuk menambahkan garis regresi yang telah dihitung sebelumnya ke dalam grafik yang sudah ada.

- perintah “polyval(p,x)” menghitung nilai prediksi berdasarkan koefisien polinomial p dan data x. Fungsi ini menggunakan nilai koefisien dari p untuk menghitung nilai y yang sesuai dengan regresi linier.





## MENGGAMBAR DAERAH YANG DIBATASI KURVA

Plot data sebenarnya berbentuk poligon. Kita juga dapat memplot kurva atau kurva terisi.

- `fillex=benar` mengisi plot.
- `style="..."`: Pilih dari “#”, “/”, “”, “/”.
- `fillcolor` : Lihat di atas untuk mengetahui warna yang tersedia.

Warna isian ditentukan oleh argumen “`fillcolor`”, dan pada <outline opsional, mencegah menggambar batas untuk semua gaya kecuali gaya default.

```
>t=linspace(0,2pi,1000); // parameter for curve
>x=sin(t)*exp(t/pi); y=cos(t)*exp(t/pi); // x(t) and y(t)
>figure(1,2); aspect(16/9)
```

```
Function figure not found.
Try list ... to find functions!
Error in:
figure(1,2); aspect(16/9) ...
      ^
```

```
>figure(1); plot2d(x,y,r=10); // plot curve
```

```
Function figure not found.
Try list ... to find functions!
Error in:
figure(1); plot2d(x,y,r=10); // plot curve ...
      ^
```

```
>figure(2); plot2d(x,y,r=10,>filled,style="/",fillcolor=red); //
fill curve
```

```
Function figure not found.
Try list ... to find functions!
Error in:
figure(2); plot2d(x,y,r=10,>filled,style="/",fillco
      ^
```

```
>figure(0):
```

```
Function figure not found.
```

```
Try list ... to find functions!
Error in:
figure(0): ...
      ^
```

Dalam contoh berikut kita memplot elips terisi dan dua segi enam terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian berbeda.

```
>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):
Function plot2d not found.
Try list ... to find functions!
Error in:
... ; plot2d(sin(x),cos(x)*0.5,r=1,&gt;filled,style="/"):
      ^

>t=linspace(0,2pi,6); ...
> plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):
Variable red not found!
Error in:
... t2d(cos(t),sin(t),&gt;filled,style="/",fillcolor=red,r=
      ^

>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):
Function plot2d not found.
Try list ... to find functions!
Error in:
... 0,2pi,6); plot2d(cos(t),sin(t),&gt;filled,style="#"):
      ^
```

Contoh lainnya adalah septagon yang kita buat dengan 7 titik pada lingkaran satuan.

```
>t=linspace(0,2pi,7); ...
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red):
Variable red not found!
Error in:
... cos(t),sin(t),r=1,&gt;filled,style="/",fillcolor=red):
      ^
```

Berikut adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah  $A[k].v \leq 3$  untuk semua baris A. Untuk mendapatkan sudut yang bagus, kita

menggunakan n yang relatif besar.

```
>A=[2,1;1,2;-1,0;0,-1];  
>function f(x,y) := max([x,y].A');  
>plot2d("f",r=4,level=[0;3],color=green,n=111):
```

Variable green not found!

Error in:

```
plot2d("f",r=4,level=[0;3],color=green,n=111): ...  
^
```

Poin utama dari bahasa matriks adalah memungkinkan pembuatan tabel fungsi dengan mudah.

```
>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);
```

Kami sekarang memiliki nilai vektor x dan y. plot2d() dapat memplot nilai-nilai ini

sebagai kurva yang menghubungkan titik-titik tersebut. Plotnya bisa diisi. Dalam hal ini

ini menghasilkan hasil yang bagus karena aturan belitan, yang digunakan untuk

isi.

```
>plot2d(x,y,<grid,<frame,>filled):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d(x,y,&lt;grid,&lt;frame,&gt;filled): ...  
^
```

Vektor interval diplot terhadap nilai x sebagai wilayah terisi antara nilai interval yang lebih rendah dan lebih tinggi.

Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tapi itu bisa

juga dapat digunakan untuk memplot kesalahan statistik.

```
>t=0:0.1:1; ...
```

```
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|");  
...
```

```
> plot2d(t,t,add=true):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
... l(t=random(size(t)),t+random(size(t))),style="|");
```

Jika  $x$  adalah vektor yang diurutkan, dan  $y$  adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi pada bidang. Gaya isiannya sama dengan gaya poligon.

```
>t=-1:0.01:1; x=t-0.01,t+0.01; y=x^3-x;
```

```
>plot2d(t,y):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d(t,y): ...
```

Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk

ini, level harus berupa matriks  $2 \times n$ . Baris pertama adalah batas bawah

dan baris kedua berisi batas atas.

```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
```

```
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y)
```

```
<= 1
```

Variable blue not found!

Error in:

```
plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y)
```

Kita juga dapat mengisi rentang nilai seperti

```
>plot2d("(x^2+y^2)^2-x^2+y^2",r=1.2,level=[-1;0],style="f"):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
... x^2+y^2)^2-x^2+y^2",r=1.2,level=[-1;0],style="f"): ...
```

```
>plot2d("cos(x)","sin(x)^3",xmin=0,xmax=2pi,>filled,style="f"):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
... x)", "sin(x)^3", xmin=0, xmax=2pi, &gt;filled, style="
```



## GRAFIK FUNGSI PARAMETRIK

Nilai  $x$  tidak perlu diurutkan.  $(x,y)$  hanya menggambarkan sebuah kurva. Jika  $x$  diurutkan, kurva tersebut merupakan grafik suatu fungsi.

Dalam contoh berikut, kita memplot spiral

Kita perlu menggunakan banyak titik untuk tampilan yang halus atau fungsi `adaptive()` untuk mengevaluasi ekspresi (lihat fungsi `adaptive()` untuk lebih jelasnya).

```
>t=linspace(0,1,1000); ...
> plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1):
Function plot2d not found.
Try list ... to find functions!
Error in:
... ,1,1000); plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1)
```

Sebagai alternatif, dimungkinkan untuk menggunakan dua ekspresi untuk kurva. Berikut ini plot kurva yang sama seperti di atas.

```
>plot2d("x*cos(2*pi*x)","x*sin(2*pi*x)",xmin=0,xmax=1,r=1):
Function plot2d not found.
Try list ... to find functions!
Error in:
... x*cos(2*pi*x) ", "x*sin(2*pi*x) ", xmin=0, xmax=1, r=1)
```

```
>t=linspace(0,1,1000); r=exp(-t); x=r*cos(2*pi*t); y=r*sin(2*pi*t);
>plot2d(x,y,r=1):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(x,y,r=1): ...
^
```

Pada contoh berikutnya, kita memplot kurvanya dengan

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t);
```

```
...
```

```
> plot2d(x,y,>filled,fillcolor=red,style="/",r=1.5):
```

```
Variable red not found!
```

```
Error in:
```

```
... (t); y=r*sin(t); plot2d(x,y,>filled,fillcolor=red,s
```

^



## MENGGAMBAR GRAFIK BILANGAN KOMPLEKS

Serangkaian bilangan kompleks juga dapat diplot. Kemudian titik-titik grid akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor garis kisi  $1 \times 2$ ) dalam argumen `cgrid`, hanya garis kisi tersebut yang terlihat.

Matriks bilangan kompleks secara otomatis akan diplot sebagai kisi-kisi pada bidang kompleks.

Pada contoh berikut, kita memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter `cgrid` menyembunyikan beberapa kurva grid.

```
>aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I*a);...  
>  
Built-in function aspect needs 1 argument (got 0)! Error in: aspect();  
r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I ... ^  
>plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):  
>aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)';  
z=r*exp(I*a);  
>plot2d(exp(z),cgrid=[40,10]):  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
plot2d(exp(z),cgrid=[40,10]): ...  
^  
  
>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);  
>plot2d(exp(z),>points,>add):  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
plot2d(exp(z),>points,>add): ...  
^
```

Vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian nyata dan bagian imajiner.

Dalam contoh, kita memplot lingkaran satuan dengan

```
>t=linspace(0,2pi,1000);...  
> plot2d(exp(I*t)+exp(4*I*t),r=2):  
Function plot2d not found.  
Try list ... to find functions!  
Error in:  
... pace(0,2pi,1000); plot2d(exp(I*t)+exp(4*I*t),r=2): ...  
^
```

## PLOT STATISTIK

Ada banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

Jumlah kumulatif dari nilai terdistribusi normal 0-1 menghasilkan jalan acak.

```
>plot2d(cumsum(randnormal(1,1000))):  
Function randnormal not found.  
Try list ... to find functions!  
Error in:  
plot2d(cumsum(randnormal(1,1000))): ...  
^
```

Penggunaan dua baris menunjukkan jalan dalam dua dimensi.

```
>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):  
Function randnormal not found.  
Try list ... to find functions!  
Error in:  
X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]): ...  
^
```

```
>columnsplot(cumsum(random(10)),style="/",color=blue):  
Variable blue not found!  
Error in:  
... lumnplot(cumsum(random(10)),style="/",color=blue
```

Itu juga dapat menampilkan string sebagai label.

```
>months=["Jan","Feb","Mar","Apr","May","Jun",...  
> "Jul","Aug","Sep","Oct","Nov","Dec"];  
>values=[10,12,12,18,22,28,30,26,22,18,12,8];  
>columnsplot(values,lab=months,color=red,style="-");  
Variable red not found!  
Error in:  
columnsplot(values,lab=months,color=red,style="-"); .  
^
```

```

>title("Temperature"):
Function title not found.
Try list ... to find functions!
Error in:
title("Temperature"): ...
      ^

>k=0:10;
>plot2d(k,bin(10,k),>bar):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(k,bin(10,k),>bar): ...
      ^

>plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add)
      ^

>plot2d(normal(1000),normal(1000),>points,grid=6,style=".."):
Function plot2d not found.
Try list ... to find functions!
Error in:
... mal(1000),normal(1000),>points,grid=6,style=".."):
      ^

>plot2d(normal(1,1000),>distribution,style="O"):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(normal(1,1000),>distribution,style="O"): ...
      ^

>plot2d("qnormal",0,5;2.5,0.5,>filled):
Function plot2d not found.
Try list ... to find functions!
Error in:

```

```
plot2d("qnormal",0,5;2.5,0.5,&gt;filled): ...
^
```

Untuk memplot distribusi statistik eksperimental, Anda dapat menggunakan `distribution=n` dengan `plot2d`.

```
>w=randexponential(1,1000); // exponential distribution
Function randexponential not found.
Try list ... to find functions!
Error in:
w=randexponential(1,1000); // exponential distribution
^
```

```
>plot2d(w,>distribution): // or distribution=n with n intervals
Variable or function w not found.
Error in:
plot2d(w,&gt;distribution): // or distribution=n with
^
```

Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan `>bar` di `plot3d`, atau dengan `plot kolom`.

```
>w=normal(1000); // 0-1-normal distribution
>{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval bounds
v
Function histo not found.
Try list ... to find functions!
Error in:
{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval
^
```

```
>plot2d(x,y,>bar):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(x,y,&gt;bar): ...
^
```

Fungsi `statplot()` mengatur gaya dengan string sederhana.

```
>statplot(1:10,cumsum(random(10)),"b"):
Function statplot not found.
Try list ... to find functions!
```

Error in:

```
statplot(1:10,cumsum(random(10)), "b"): ...  
^
```

```
>n=10; i=0:n; ...
```

```
> plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); ...
```

```
> plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
... i=0:n; plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); plo  
^
```

Selain itu, data dapat diplot sebagai batang. Dalam hal ini, x harus diurutkan dan satu elemen lebih panjang dari y. Batangnya akan memanjang dari  $x[i]$  hingga  $x[i+1]$  dengan nilai  $y[i]$ . Jika x berukuran sama dengan y, maka x akan diperpanjang satu elemen dengan spasi terakhir.

Gaya isian dapat digunakan seperti di atas.

```
>n=10; k=bin(n,0:n); ...
```

```
> plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```

Variable lightgray not found!

Error in:

```
... plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray): ..  
^
```

Data untuk plot bar ( $\text{bar}=1$ ) dan histogram ( $\text{histogram}=1$ ) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan  $>\text{distribusi}$  (atau  $\text{distribusi}=n$ ). Histogram nilai xv akan dihitung secara otomatis dengan  $>\text{histogram}$ . Jika  $>\text{even}$  ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

```
>plot2d(normal(10000),distribution=50):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d(normal(10000),distribution=50): ...  
^
```

```
>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
... 0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,&gt;b
```

```
>columnsplo(m,k):
```

```
Function columnsplo not found.
Try list ... to find functions!
Error in:
columnsplo(m,k): ...
^
```

```
>plot2d(random(600)*6,histogram=6):
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(random(600)*6,histogram=6): ...
^
```

Untuk distribusi, terdapat parameter `distribution=n`, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan `n` sub-interval.

```
>plot2d(normal(1,1000),distribution=10,style="\"): 
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(normal(1,1000),distribution=10,style="\"): ..
^
```

Dengan parameter `even=true`, ini akan menggunakan interval bilangan bulat.

```
>plot2d(intrandom(1,1000,10),distribution=10,even=true):
```

```
Function plot2d not found.
Try list ... to find functions!
Error in:
... d(intrandom(1,1000,10),distribution=10,even=true)
```

Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Silahkan lihat tutorial tentang statistik.

```

>columnspot(getmultiplicities(1:6,intrandom(1,6000,6))):
Function getmultiplicities not found.
Try list ... to find functions!
Error in:
... nsplot(getmultiplicities(1:6,intrandom(1,6000,6))): ...
^

>plot2d(normal(1,1000),>distribution);...
> plot2d("qnormal(x)",color=red,thickness=2,>add):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(normal(1,1000),>distribution);    plot2d("qnormal
^

```

Ada juga banyak plot khusus untuk statistik. Plot kotak menunjukkan kuartil distribusi ini dan banyak outlier. Menurut definisinya, outlier dalam plot kotak adalah data yang melebihi 1,5 kali rentang 50% tengah plot.

```

>M=normal(5,1000); boxplot(quartiles(M)):
Function quartiles not found.
Try list ... to find functions!
Error in:
M=normal(5,1000); boxplot(quartiles(M)): ...
^

```



## FUNGSI IMPLISIT

Fungsi implisit adalah jenis fungsi dimana variabel dependen tidak dapat dipisahkan secara eksplisit dari variabel independen. Bentuk dari fungsi implisit sendiri adalah  $f(x,y)=0$ , dimana variabel, koefisien, dan konstanta sebagai persamaan di sisi kiri, dan disamakan dengan nol.

Contoh dari fungsi implisit misalnya persamaan lingkaran dan persamaan elips.

Plot implisit menunjukkan penyelesaian garis level  $f(x,y)=\text{level}$ , dengan “level” dapat berupa nilai tunggal atau vektor nilai. Jika level = “auto”, akan ada garis level  $n_c$ , yang akan tersebar antara fungsi minimum dan maksimum secara merata.

Perintah `>hue` untuk membuat Warna yang lebih gelap atau lebih terang untuk menunjukkan nilai fungsi. Untuk fungsi implisit, `xv` harus berupa fungsi atau ekspresi parameter `x` dan `y`, atau alternatifnya, `xv` dapat berupa matriks nilai.

Euler dapat menandai garis level  
dari fungsi apa pun.

Untuk menggambar himpunan  $f(x,y)=c$  untuk satu atau lebih konstanta `c`, Anda dapat menggunakan `plot2d()` dengan plot implisitnya pada bidang. Parameter `c` adalah `level=c`, dimana `c` dapat berupa vektor garis level. Selain itu, skema warna dapat digambar di latar belakang untuk menunjukkan nilai fungsi setiap titik dalam plot. Parameter “`n`” menentukan kehalusan plot.

```
>aspect();
```

```
Built-in function aspect needs 1 argument (got 0)!  
Error in:  
aspect(); ...  
^
```

```
>plot2d("x^2+y^2-x*y-x",level=0,r=1.5,contourcolor=red):
```

```
Variable red not found!
```

```
Error in:
```

```
... 2d("x^2+y^2-x*y-x",level=0,r=1.5,contourcolor=red)
```

```

>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=0): // Solutions of f(x,y)=0
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(expr,level=0): // Solutions of f(x,y)=0 ...
^

>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=10):
// nice
Variable white not found!
Error in:
... lot2d(expr,level=0:0.5:20,&gt;hue,contourcolor=white,n=10):
^

>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4): //
nicer
Function plot2d not found.
Try list ... to find functions!
Error in:
... (expr,level=0:0.5:20,&gt;hue,&gt;spectral,n=200,grid=4):
^

Ini juga berfungsi untuk plot data. Namun Anda harus
menentukan rentangnya untuk label sumbu.

>x=-2:0.05:1; y=x'; z=expr(x,y);
>plot2d(z,level=0,a=-1,b=2,c=-2,d=1,>hue):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(z,level=0,a=-1,b=2,c=-2,d=1,&gt;hue): ...
^

>plot2d("x^3-y^2",>hue,>spectral,>contour):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("x^3-y^2",&gt;hue,&gt;spectral,&gt;contour): ...

```

Perintah `contour` adalah untuk menentukan bahwa grafik yang dihasilkan adalah grafik kontur. grafik kontur menunjukkan garis-garis yang menghubungkan titik-titik dengan nilai fungsi yang sama, sehingga membuat visualisasi bentuk permukaan fungsi.

```
>plot2d("x^3-y^2",level=0,contourwidth=3,contourcolor=red,>add):
Variable red not found!
Error in:
... "x^3-y^2",level=0,contourwidth=3,contourcolor=red,
>z=z+normal(size(z))*0.9;
>plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1): ...
```

Perintah ini untuk menambahkan noise atau gangguan acak pada fungsi `z`. nilai yang dikalikan adalah standar deviasi atau besaran gangguan yang diberikan. semakin besar nilai yang dimasukkan maka semakin besar pula gangguan pada fungsi.

```
>plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):
Variable lightgray not found!
Error in:
... xpr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray,
>plot2d("x^2+y^3+x*y",level=1,r=4,n=100):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("x^2+y^3+x*y",level=1,r=4,n=100): ...
```

```
>plot2d("x^2+2*y^2-x*y",level=0:0.1:10,n=100,contourcolor=white,>hue):
Variable white not found!
Error in:
```

```
... *y^2-x*y", level=0:0.1:10, n=100, contourcolor=white, &gt;
```

Dimungkinkan juga untuk mengisi set dengan rentang level.

Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

```
>plot2d(expr,level=[0;1],style="",color=blue): // 0 <= f(x,y)
<= 1
Variable blue not found!
Error in:
plot2d(expr,level=[0;1],style="",color=blue): // 0 &lt;= f
```

Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks interval level 2xn, di mana baris pertama berisi awal dan baris kedua berisi akhir setiap interval. Alternatifnya, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

```
>plot2d("x^4+y^4",level=[0;1],r=1.5,color=blue,style="/"):
Variable blue not found!
Error in:
plot2d("x^4+y^4",level=[0;1],r=1.5,color=blue,style="/"):
```

```
>plot2d("x^2+y^3+x*y",level=[0,2,4;1,3,5],style="/",r=2,n=100):
Function plot2d not found.
Try list ... to find functions!
Error in:
... y^3+x*y", level=[0,2,4;1,3,5], style="/", r=2, n=100): ...
```

```
>plot2d("x^2+y^3+x*y",level=-10:20,dl=0.1,r=5,style="-",n=100):
Function plot2d not found.
Try list ... to find functions!
Error in:
... y^3+x*y", level=-10:20, dl=0.1, r=5, style="-", n=100): ...
```

```

>plot2d("sin(x)*cos(y)",>levels,r=2*pi,>hue,n=100):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("sin(x)*cos(y)",>levels,r=2*pi,>hue,n=100):

```

Dimungkinkan juga untuk menandai suatu wilayah

Hal ini dilakukan dengan menambahkan level dengan dua baris.

```

>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=6,...
> style="color=red,<outline,...
> level=[-2;0],n=100):
Variable red not found!
Error in:
... "(x^2+y^2-1)^3-x^2*y^3",r=6, style="",color=red,

```

```

>plot2d("4x^2-4.5",-0.75,0.75,thickness=3,>add);plot2d([0,0],[3,5],thic
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("4x^2-4.5",-0.75,0.75,thickness=3,>add);plot2d([0,0],[3,5],thic

```

```

>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=6,...
> style="color=red,<outline,...
> level=[-1;0],n=100):
Variable red not found!
Error in:
... "(x^2+y^2-1)^3-x^2*y^3",r=6, style="",color=red,

```

```

>plot2d([-3.5,-3.5],[-1.1,1.1],thickness=3,>add);plot2d("4*(x-
4)^2-1",3.3,4.7,thickness=3,>add):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d([-3.5,-3.5],[-1.1,1.1],thickness=3,>add);plot2d("4*(x-

```

>

Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi persamaan seperti

```
>plot2d("x^3-x*y+x^2*y^2",r=6,level=1,n=100):
```

Function plot2d not found.

Try list ... to find functions!

Error in:

```
plot2d("x^3-x*y+x^2*y^2",r=6,level=1,n=100): ...  
^
```

```
>function starplot1 (v, style="/", color=green, lab=none)...
```

>

Variable green not found!

Error in:

```
function starplot1 (v, style="/", color=green, lab=none)..  
^
```

Tidak ada tanda centang kotak atau sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plotnya.

Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Ini tidak perlu dilakukan jika Anda yakin plot Anda berhasil.

```
>reset; starplot1(normal(1,10)+2,color=red,lab=1:10):
```

Variable reset not found!

Error in:

```
reset; starplot1(normal(1,10)+2,color=red,lab=1:10): ...  
^
```

- Starplot1 adalah perintah untuk menggambar plot berbentuk bintang.
- Ini adalah fungsi untuk menghasilkan vektor acak yang mengikuti
- distribusi normal dengan rata-rata 1 dan deviasi standar 10. Fungsi
- ini menghasilkan data yang terdistribusi normal dengan nilai-nilai
- acak.
- +5 adalah untuk menambahkan 5 ke setiap elemen dari vektor yang

- terdistribusi normal.

Terkadang, Anda mungkin ingin merencanakan sesuatu yang `plot2d` tidak bisa lakukan, tapi hampir.

Dalam fungsi berikut, kita membuat plot impuls logaritmik. `plot2d` dapat melakukan plot logaritmik, tetapi tidak untuk batang impuls.

```
>function logimpulseplot1 (x,y) ...
{ x0,y0}=makeimpulse (x, log (y) /log (10) );
plot2d(x0,y0,>bar,grid=0);
h=holding(1);
frame();
xgrid(ticks(x));
p=plot();
for i=-10 to 10;
    if i<=p[4] and i>=p[3] then
        ygrid(i,yt="10^"+i);
    endif;
end;
holding(h);
endfunction
```

Mari kita uji dengan nilai yang terdistribusi secara eksponensial.

```
>aspect(); x=1:10; y=-log(random(size(x)))*200; ...
> logimpulseplot1(x,y):
Built-in function aspect needs 1 argument (got 0)!
Error in:
aspect(); x=1:10; y=-log(random(size(x)))*200; logimp
```

Mari kita menganimasikan kurva 2D menggunakan `plot` langsung. Perintah `plot(x,y)` hanya memplot kurva ke dalam jendela `plot`. `setplot(a,b,c,d)` menyetel jendela ini.

Fungsi `wait(0)` memaksa plot muncul di jendela grafis. Jika tidak, pengundian ulang akan dilakukan dalam interval waktu yang jarang.

```
>function animliss (n,m) ...
t=linspace(0,2pi,500);
f=0;
```

```

c=framecolor(0);
l=linewidth(2);
setplot(-1,1,-1,1);
repeat
    clg;
    plot(sin(n*t),cos(m*t+f));
    wait(0);
    if testkey() then break; endif;
    f=f+0.02;
end;
framecolor(c);
linewidth(l);
endfunction

```

Tekan tombol apa saja untuk menghentikan animasi ini.

>animliss(2,3); // lihat hasilnya, jika sudah puas, tekan ENTER

Built-in function setplot needs 1 argument (got 4)!

Try "trace errors" to inspect local variables after errors

animliss:

```

    setplot(-1,1,-1,1);

```



## PLOT LOGARITMIK

EMT menggunakan parameter “logplot” untuk skala logaritmik.

Plot logaritma dapat diplot menggunakan skala logaritma di y dengan logplot=1, atau menggunakan skala logaritma di x dan y dengan logplot=2, atau di x dengan logplot=3.

- logplot=1: y-logarithmic
- logplot=2: x-y-logarithmic
- logplot=3: x-logarithmic

```
> plot2d("exp(x3-x*x2)",1,5,logplot=1):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("exp(x3-x*x2)",1,5,logplot=1): ...  
^
```

```
> plot2d("exp(x+sin(x))",0,100,logplot=1):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("exp(x+sin(x))",0,100,logplot=1): ...  
^
```

```
> plot2d("exp(x+sin(x))",10,100,logplot=2):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("exp(x+sin(x))",10,100,logplot=2): ...  
^
```

```
> plot2d("gamma(x)",1,10,logplot=1):
```

```
Function plot2d not found.
```

```
Try list ... to find functions!
```

```
Error in:
```

```
plot2d("gamma(x)",1,10,logplot=1): ...
```

```

^
>plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3): ...
^

```

Ini juga berfungsi dengan plot data.

```

>x=10^(1:20); y=x^2-x;
>plot2d(x,y,logplot=2):
Function plot2d not found.
Try list ... to find functions!
Error in:
plot2d(x,y,logplot=2): ...
^

```

## RUJUKAN LENGKAP FUNGSI PLOT2D()

function plot2d (xv, yv, btest, a, b, c, d, xmin, xmax, r, n, ..  
logplot, grid, frame, framecolor, square, color, thickness, style, ..  
auto, add, user, delta, points, addpoints, pointstyle, bar, histogram, ..  
distribution, even, steps, own, adaptive, hue, level, contour, ..  
nc, filled, fillcolor, outline, title, xl, yl, maps, contourcolor, ..  
contourwidth, ticks, margin, clipping, cx, cy, insimg, spectral, ..  
cgrid, vertical, smaller, dl, niveau, levels)

Multipurpose plot function for plots in the plane (2D plots). This function can do plots of functions of one variables, data plots, curves in the plane, bar plots, grids of complex numbers, and implicit plots of functions of two variables.

### Parameters

x,y : equations, functions or data vectors

a,b,c,d : Plot area (default a=-2,b=2)

r : if r is set, then a=cx-r, b=cx+r, c=cy-r, d=cy+r

r can be a vector [rx,ry] or a vector [rx1,rx2,ry1,ry2]

xmin,xmax : range of the parameter for curves

auto : Determine y-range automatically (default)

square : if true, try to keep square x-y-ranges

n : number of intervals (default is adaptive)

grid : 0 = no grid and labels,

1 = axis only,

2 = normal grid (see below for the number of cells)

3 = inside axis

4 = no grid

5 = full grid including margin

6 = ticks at the frame

7 = axis only

8 = axis only, sub-ticks

**frame : 0 = no frame**

**framecolor:** color of the frame and the grid

**margin :** number between 0 and 0.4 for the margin around the

**plot**

**color :** Color of curves. If this is a vector of colors,

it will be used for each row of a matrix of plots.

point plots, it should be a column vector. If a row

full matrix of colors is used for point plots, it

each data point.

**thickness :** line thickness for curves

This value can be smaller than 1 for very thin lines

**style :** Plot style for lines, markers, and fills.

For points use

"[]", "<>", ".", "..", "...",

"\*", "+", "|", "-", "o"

"[]#", "&lt;&gt;#", "o#" (filled shapes)

"[]w", "&lt;&gt;w", "ow" (non-transparent)

For lines use

"-", "--", "-.", ".", ".-.", "-.-", "-&gt;"

For filled polygons or bar plots use

"#", "#O", "O", "/", "\", "\/",

"+", "|", "-", "t"

points : plot single points instead of line segments

addpoints : if true, plots line segments and points

add : add the plot to the existing plot

user : enable user interaction for functions

delta : step size for user interaction

bar : bar plot (x are the interval bounds, y the interval values)

histogram : plots the frequencies of x in n subintervals

distribution=n : plots the distribution of x with n subintervals

even : use inter values for automatic histograms.

steps : plots the function as a step function (steps=1,2)

adaptive : use adaptive plots (n is the minimal number of steps)

level : plot level lines of an implicit function of two variables

outline : draws boundary of level ranges.

If the level value is a 2xn matrix, ranges of levels will be drawn in the color using the given fill style. If outline is true, it will be drawn in the contour color. Using this feature, regions of  $f(x,y)$  between limits can be marked.

hue : add hue color to the level plot to indicate the function

value

contour : Use level plot with automatic levels

nc : number of automatic level lines

title : plot title (default “”)

xl, yl : labels for the x- and y-axis

smaller : if >0, there will be more space to the left for labels.

vertical :

Turns vertical labels on or off. This changes the global variable `vertical` locally for one plot. The value 1 sets only vertical text, the value 2 uses vertical numerical labels on the y axis.

filled : fill the plot of a curve

fillcolor : fill color for bar and filled curves

outline : boundary for filled polygons

logplot : set logarithmic plots

1 = logplot in y,

2 = logplot in xy,

3 = logplot in x

own :

A string, which points to an own plot routine. With >user, you get

the same user interaction as in `plot2d`. The range will be set before each call to your function.

maps : map expressions (0 is faster), functions are always mapped.

contourcolor : color of contour lines

contourwidth : width of contour lines

clipping : toggles the clipping (default is true)

title :

This can be used to describe the plot. The title will appear above the plot. Moreover, a label for the x and y axis can be added with `xl="string"` or `yl="string"`. Other labels can be added with the functions `label()` or `labelbox()`. The title can be a unicode string or an image of a Latex formula.

cgrid :

Determines the number of grid lines for plots of complex grids. Should be a divisor of the the matrix size minus 1 (number of subintervals). `cgrid` can be a vector `[cx,cy]`.

Overview

The function can plot

- expressions, call collections or functions of one variable,
- parametric curves,
- x data against y data,
- implicit functions,
- bar plots,
- complex grids,
- polygons.

If a function or expression for `xv` is given, `plot2d()` will compute values in the given range using the function or expression. The expression must be an expression in the variable `x`. The range must

be defined in the parameters `a` and `b` unless the default range `[-2,2]` should be used. The `y`-range will be computed automatically,

unless `c` and `d` are specified, or a radius `r`, which yields the range `[-r,r]` for `x` and `y`. For plots of functions, `plot2d` will use an adaptive evaluation of the function by default. To speed up the

plot for complicated functions, switch this off with `<adaptive`,  
and

optionally decrease the number of intervals `n`. Moreover,  
`plot2d()`

will by default use mapping. I.e., it will compute the plot  
element

for element. If your expression or your functions can handle a  
vector `x`, you can switch that off with `<maps` for faster  
evaluation.

Note that adaptive plots are always computed element for  
element.

If functions or expressions for both `xv` and for `yv` are specified,  
`plot2d()` will compute a curve with the `xv` values as x-  
coordinates

and the `yv` values as y-coordinates. In this case, a range should  
be

defined for the parameter using `xmin`, `xmax`. Expressions  
contained

in strings must always be expressions in the parameter variable  
`x`.