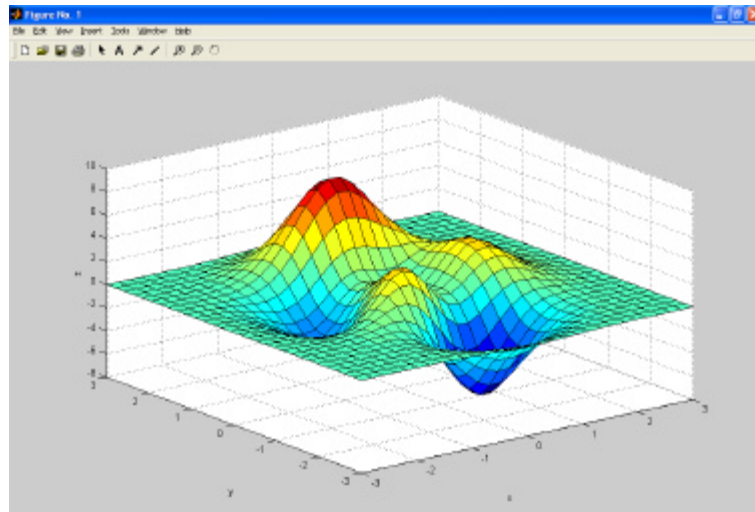


Usando o comando FMINSEARCH – Mínimos Locais e Globais:

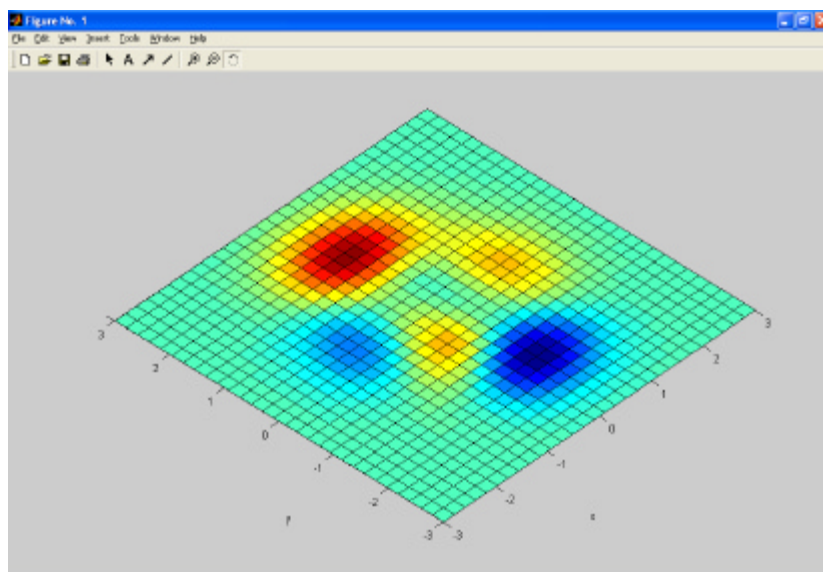
Para testar a função de minimização do Matlab vamos usar uma função em três dimensões. Ou seja, dado X e Y calcula Z. A função da figura abaixo é um exemplo disponibilizado pelo Matlab, chamado “peaks”. Para gerar a figura abaixo uso o código:

```
[X,Y,Z] = peaks(30);  
surf(X,Y,Z)  
xlabel('x')  
ylabel('y')  
zlabel('z')
```

A primeira linha gera matrizes X e Y de dimensão 30 x 30, e calcula Z. O resultado é graficado via comando SURF:



Observe a presença de pelo menos dois mínimos, em azul:



Vamos criar uma função nossa que, dado X e Y, calcule Z. Ou seja:

```
function [e] = minha(w)
e=peaks(w(1),w(2));
```

Essa função recebe uma matriz de dimensão dois, chamada W. Onde o primeiro argumento é X e o segundo é Y. E a função retorna “e”, ou seja, Z. Testando:

```
>> [e] = minha([0 1])
```

```
e = 3.6886
```

Fazendo X=0 e Y=1, temos Z=3.6886.

Observe que preciso arranjar uma combinação de valores X e Y que minimize Z. Testemos algumas combinações:

```
>> [e] = minha([1 1])
```

```
e = 2.4338
```

```
>> [e] = minha([1 0])
```

```
e = 2.9369
```

```
>> [e] = minha([0 0])
```

```
e = 0.9810
```

Testamos mais 3 chutes e o melhor resultado foi um Z de 0.9810. Preciso ficar um bom tempo testando valores de X e Y até encontrar algum mínimo em Z.... Como fazer isso de modo mais inteligente??

Ai entra a função FMINSEARCH:

```
[respostaxy, valorz] = fminsearch('minha',[0 1])
```

O “fminsearch” tem como:

- primeiro argumento a função que quero minimizar (‘minha’)
- segundo argumento o chute inicial: [x0 y0] ou seja: [0 1]
- primeiro parâmetro de saída (respostaxy)
- segundo parâmetro de saída (valorz)

Com o chute inicial $X=0$ e $Y=1$, o `fminsearch` caminha para:

```
respostaxy = 0.2965 0.3202
valorz = -0.0649
```

Tentemos outro chute inicial:

```
>> [respostaxy, valorz] = fminsearch('minha',[2 -2])
```

```
respostaxy = 0.2283 -1.6255
valorz = -6.5511
```

Esse valor de Z é bem menor! Logo o primeiro era um mínimo local e esse tem alguma chance de ser um mínimo global.

Uma observação final: é possível mostrar todo o percurso que o método usou desde o chute inicial até convergir. Regule a propriedade “display” para “iter” através do comando “optimset”:

```
options = optimset('Display','iter');
[respostaxy, valorz] = fminsearch('minha',[0 1],options)
```

O resultado é:

Iteration	Func-count	min f(x)	Procedure
1	3	3.68844	initial
2	5	2.65309	expand
3	7	2.19172	expand
4	9	0.773891	expand
5	11	0.398715	expand
6	12	0.398715	reflect
7	14	0.398715	contract inside
8	16	0.398715	contract inside
9	18	0.398715	contract inside
10	20	0.398715	contract inside
11	22	0.398715	contract inside
12	24	0.398715	contract outside
13	26	0.398687	contract inside
14	28	0.398596	reflect
15	30	0.3983	reflect
16	31	0.3983	reflect
17	33	0.397947	reflect
18	35	0.397947	contract inside
19	37	0.397824	reflect
20	39	0.397521	expand

21	41	0.397291	expand
22	43	0.39622	expand
23	45	0.395983	expand
24	47	0.392511	expand
25	48	0.392511	reflect
26	50	0.385904	expand
27	51	0.385904	reflect
28	53	0.378757	expand
29	55	0.366545	expand
30	57	0.352373	expand
31	59	0.31384	expand
32	61	0.291091	expand
33	63	0.195469	expand
34	64	0.195469	reflect
35	66	0.0716163	expand
36	68	-0.00848879	expand
37	69	-0.00848879	reflect
38	70	-0.00848879	reflect
39	72	-0.043388	reflect
40	74	-0.0463668	contract inside
41	76	-0.0629303	reflect
42	78	-0.0629303	contract inside
43	80	-0.0629303	contract inside
44	82	-0.0637175	contract inside
45	84	-0.0640829	contract inside
46	86	-0.0648699	contract inside
47	87	-0.0648699	reflect
48	89	-0.0648699	contract inside
49	91	-0.0648699	contract outside
50	93	-0.0649086	contract inside
51	95	-0.0649217	contract inside
52	97	-0.0649295	contract inside
53	99	-0.0649311	contract inside
54	101	-0.0649332	contract outside
55	103	-0.064935	contract inside
56	105	-0.0649354	contract inside
57	107	-0.0649357	contract outside
58	109	-0.0649358	contract inside
59	111	-0.0649358	contract inside
60	113	-0.0649358	contract inside
61	115	-0.0649359	contract inside
62	117	-0.0649359	contract inside
63	119	-0.0649359	contract outside

Optimization terminated successfully:

the current x satisfies the termination criteria using `OPTIONS.TolX` of $1.000000e-004$
and $F(X)$ satisfies the convergence criteria using `OPTIONS.TolFun` of $1.000000e-004$