

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
```

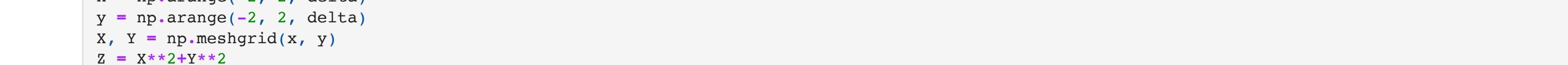
```
In [2]: def f(x):
# x numpy array
s=x[0]**2+x[1]**2
return s
```

```
In [ ]:
```

```
In [3]: import matplotlib.cm as cm
# βιαια το πλεγμα
delta = 0.025
x = np.arange(-2, 2, delta)
y = np.arange(-2, 2, delta)
X, Y = np.meshgrid(x, y)
Z = X**2+Y**2
```

```
In [5]: #Ζωγραφίζω τις ισοαθμικες
fig, ax = plt.subplots()
CS = ax.contour(X, Y, Z, levels=[2,5])
ax.clabel(CS, inline=True, fontsize=10)
ax.set_title('Simplest default with labels')
```

Out[5]: Text(0.5, 1.0, 'Simplest default with labels')



```
In [6]: def fl(x):
s=np.sqrt(1+x[0]**2)+np.sqrt(1+x[1]**2)
return s
```

```
In [7]: Z = np.sqrt(1+X**2)+np.sqrt(1+Y**2)
#Ζωγραφίζω τις ισοαθμικες
fig, ax = plt.subplots()
CS = ax.contour(X, Y, Z)
ax.clabel(CS, inline=True, fontsize=10)
ax.set_title('Simplest default with labels')
```

Out[7]: Text(0.5, 1.0, 'Simplest default with labels')



```
In [8]: # Κρισιμα σημεια
# grad_f=0
```

```
In [9]: def grad_f(x):
# x numpy array
s1=2*x[0]
s2=2*x[1]
s=np.array([s1,s2])
return s
```

```
In [10]: def hessian_f(x):
# x numpy array
s11=2.
s12=0
s21=0
s22=2.
s=np.array([[s11,s12],[s21,s22]])
return s
```

```
In [11]: x=np.array([0,0])
print(grad_f(x))
```

[0 0]

```
In [12]: print(hessian_f(x))
```

[[2. 0.]
[0. 2.]]

```
In [13]: ##Μεθοδος Νευτωνα
x0=np.array([1,1])
A=hessian_f(x0)
b=-grad_f(x0)
##solve Ap=b
p=np.linalg.solve(A,b)
print(p)
x1=x0+p
print(x1)
```

[-1. -1.]
[0. 0.]

```
In [14]: def grad_fl(x):
s1=-x[0]/np.sqrt(1+x[0]**2)
s2=-x[1]/np.sqrt(1+x[1]**2)
s=np.array([s1,s2])
return s
```

```
In [15]: x=np.array([0,0])
print(grad_fl(x))
```

[0. 0.]

```
In [16]: def hessian_fl(x):
s11=1./(1+x[0]**2)**(3./2)
s12=0.
s21=0.
s22=1./(1+x[1]**2)**(3./2)
s=np.array([[s11, s12],[s21,s22]])
return s
```

```
In [17]: import numpy as np
def is_pos_def(A):
return np.all(np.linalg.eigvals(A) > 0)
```

```
In [18]: ##Μεθοδος Νευτωνα
x0=np.array([1,1])
err=np.linalg.norm(grad_f(x0))
tol=1.e-10
iter=0
while (err>tol) and iter<10:
A=hessian_f(x0)
print('Θετικα ορισμενος:', is_pos_def(A))
print('Δεικτης καταστασης:', np.linalg.cond(A))
b=-grad_f(x0)
##solve Ap=b
p=np.linalg.solve(A,b)
x1=x0+p
print(x1)
err=np.linalg.norm(grad_f(x1))
print('grad=', err)
x0=x1
iter=iter+1
print('-----')
```

print(iter)
print(x1)

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.0
[3. 3.]
grad= 1.3416407864998738

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.0
[33. 33.]
grad= 1.4135646916977938

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.0
[36003. 36003.]
grad= 1.414213561827579

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.0
[4.6667665e+13 4.6667665e+13]
grad= 1.4142135623730951

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.0
[1.01636153e+41 1.01636153e+41]
grad= 1.4142135623730951

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.0
[1.04989205e+123 1.04989205e+123]
grad= 1.4142135623730951

Θετικα ορισμενος: False
Δεικτης καταστασης: inf

/var/folders/k1/09d19vm1vn5ff_c3ddmxc7m0000gn/T/ipykernel_70235/2212585952.py:2: RuntimeWarning: overflow encountered in double_scalars
s1=1./(1+x[0]**2)**(3./2)
/var/folders/k1/09d19vm1vn5ff_c3ddmxc7m0000gn/T/ipykernel_70235/2212585952.py:5: RuntimeWarning: overflow encountered in double_scalars
s22=1./(1+x[1]**2)**(3./2)

LinAlgError Traceback (most recent call last)
/var/folders/k1/09d19vm1vn5ff_c3ddmxc7m0000gn/T/ipykernel_70235/3918498271.py in <module>
10 b=-grad_fl(x0)
11 ##solve Ap=b
--> 12 p=np.linalg.solve(A,b)
13 x1=x0+p
14 print(x1)

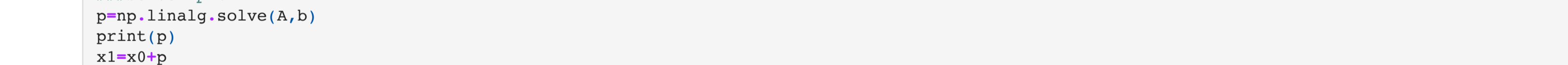
<_array_function__ internals> in solve(*args, **kwargs)
~/opt/anaconda3/lib/python3.9/site-packages/numpy/linalg/linalg.py in solve(a, b)
391 signature = 'DD->D' if isComplexType(t) else 'dd->d'
392 extobj = get_linalg_error_extobj(_raise_linalgerror_singular)
--> 393 r = gufunc(a, b, signature=signature, extobj=extobj)
394
395 return wrap(r.astype(result_t, copy=False))
~/opt/anaconda3/lib/python3.9/site-packages/numpy/linalg/linalg.py in _raise_linalgerror_singular(err, flag)
86
87 def _raise_linalgerror_singular(err, flag):
--> 88 raise LinAlgError("Singular matrix")
89
90 def _raise_linalgerror_nonposdef(err, flag):

LinAlgError: Singular matrix

```
In [19]: def rosenbrock(x):
s=100*(x[1]-x[0]**2)**2+(1-x[0])**2
return s
```

```
In [20]: Z = 100*(Y-X**2)**2+(1-X)**2
#Ζωγραφίζω τις ισοαθμικες
fig, ax = plt.subplots()
CS = ax.contour(X, Y, Z, levels=[1,10,50,100])
ax.clabel(CS, inline=True, fontsize=10)
ax.set_title('Simplest default with labels')
```

Out[20]: Text(0.5, 1.0, 'Simplest default with labels')



```
In [21]: def grad_r(x):
s1=-2*(1-x[0])
s2=200*(x[1]-x[0]**2)*x[0]
s=np.array([s1,s2])
return s
```

```
In [22]: def hessian_r(x):
s11=2-400*(x[1]-x[0])**800*x[0]
s12=-400*x[0]
s21=-400*x[0]
s22=200
s=np.array([[s11, s12],[s21,s22]])
return s
```

```
In [23]: x=np.array([1,1])
print(grad_r(x))
```

[0 0]

```
In [24]: ##Μεθοδος Νευτωνα
x0=np.array([-0.1,-1])
err=np.linalg.norm(grad_r(x0))
tol=1.e-10
iter=0
while (err>tol) and iter<10:
A=hessian_r(x0)
print('Θετικα ορισμενος:', is_pos_def(A))
print('Δεικτης καταστασης:', np.linalg.cond(A))
b=-grad_f(x0)
##solve Ap=b
p=np.linalg.solve(A,b)
x1=x0+p
print(x1)
err=np.linalg.norm(grad_r(x1))
print('grad=', err)
x0=x1
iter=iter+1
print('-----')
```

print(iter)
print(x1)

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.62355778077423
[-0.09984702 -1.00356613]
grad= 207.15134114063775

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6284819108952868
[-0.09969427 -1.00713846]
grad= 207.86080195291788

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6334893901695795
[-0.09954175 -1.01071695]
grad= 208.57149387501352

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6385781524710785
[-0.09938949 -1.01430159]
grad= 209.28341268125325

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6437461788179117
[-0.09923746 -1.01789236]
grad= 209.9965541343572

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.648991503029561
[-0.09908569 -1.02148922]
grad= 210.7109139857938

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6543122110074553
[-0.09893417 -1.02509217]
grad= 211.42648797645282

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6597064399510066
[-0.09878291 -1.02870116]
grad= 212.14327183688496

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6651723775173588
[-0.0986319 -1.03231619]
grad= 212.86126128785736

Θετικα ορισμενος: True
Δεικτης καταστασης: 1.6707082609326454
[-0.09848117 -1.03593722]
grad= 213.58045204077607

10
[-0.09848117 -1.03593722]

```
In [ ]:
```

```
In [ ]:
```