W4

April 2, 2020

1 Worksheet 4 Unconstrained optimization

1.1 J.Morlier March 2020

2 On the descent direction

[34]:
```
using LinearAlgebra, SparseArrays, SuiteSparse, Random
using SymPy
using Plots; pyplot()
import Base.print_matrix
```

[35]:
```
versioninfo()
```

```
Julia Version 1.3.1
Commit 2d5741174c (2019-12-30 21:36 UTC)
Platform Info:
  OS: macOS (x86_64-apple-darwin18.6.0)
  CPU: Intel(R) Core(TM) i5-8259U CPU @ 2.30GHz
  WORD_SIZE: 64
  LIBM: libopenlibm
  LLVM: libLLVM-6.0.1 (ORCJIT, skylake)
```

[36]:
```
entries = @syms x1 x2  real=true
```

[36]:
```
(x1, x2)
```

Consider the function f : R2 → R given by J(x) = x1^{3+x}2 +2*x2. Give some descent directions for J at the point [1,1] . Is there any descent direction at the point $[0, -1]$? Justify the answer.

[132]:
```
J=x1^3 +x2^2 +2*x2
```

[132]:

$$x_1^3 + x_2^2 + 2x_2$$

[133]:
```
dJdx1=diff(J,x1)  ;

dJdx2=diff(J,x2) ;
```

```
dJ=[dJdx1; dJdx2]
```

[133]:

$$\begin{bmatrix} 3x_1^2 \\ 2x_2 + 2 \end{bmatrix}$$

substitute with Point (1,1);

[134]:
```
A=[dJdx1(1) ; dJdx2(1)]
```

[134]:

$$\begin{bmatrix} 3 \\ 4 \end{bmatrix}$$

Create vector of possible descent directions

[135]:
```
entries = @syms d1 d2  real=true
```

[135]: (d1, d2)

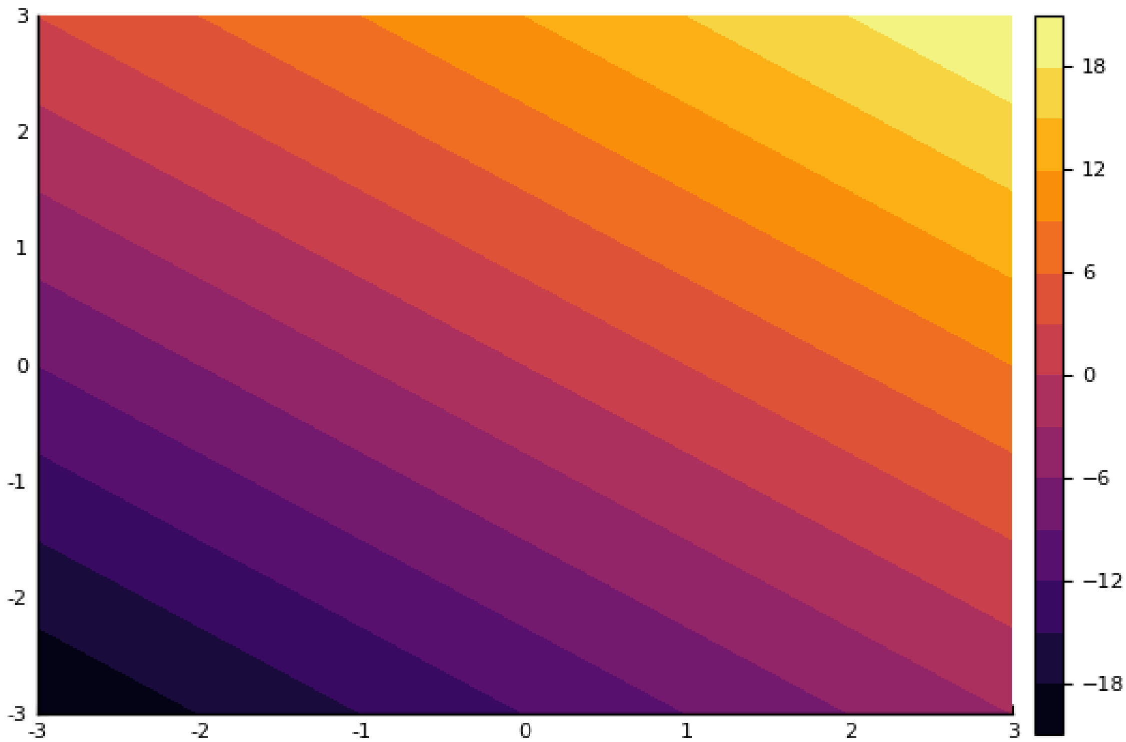
[136]:
```
dir=A'*[d1; d2]
```

[136]:

$$3d_1 + 4d_2$$

[137]:
```
x=range(-3,stop=3,length=20)
y=range(-3,stop=3,length=20)
f(x,y) = 3*x+4*y
plot(x,y,f,st=:contourf,camera=(0,0))
```

[137]:



It's up to you to choose one possible solution such that 3d1 + 4d2 < 0

substitute with (0,-1). Something wrong?

[142]:
```
dJdx1
```

[142]:

$$3x_1^2$$

[143]:
```
dJdx2
```

[143]:

$$2x_2 + 2$$

[144]:
```
A=[dJdx1(0) ; dJdx2(-1)]
```

[144]:

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

[145]:
```
dir=A'*[d1; d2]
```

[145]:

$$0$$

thus one cannot find a descent direction at the point.

3 On the optimal step-size

Compute the optimal step-size at [5, 1] following the direction d = − f (5, 1).

[146]:
```
J=0.5*x1^2 + 4.5*x2^2
```

[146]:

$$0.5x_1^2 + 4.5x_2^2$$

[147]:
```
dJdx1=diff(J,x1)  ;

dJdx2=diff(J,x2) ;

dJ=[dJdx1; dJdx2]
```

[147]:

$$\begin{bmatrix} 1.0x_1 \\ 9.0x_2 \end{bmatrix}$$

[148]:
```
entries = @syms alpha  real=true
```

[148]: (alpha,)

[149]:
```
d=-dJ
```

[149]:

$$\begin{bmatrix} -1.0x_1 \\ -9.0x_2 \end{bmatrix}$$

[150]:
```
x=[x1;x2]
```

[150]:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

[152]:
```
Jnew=(0.5*newx[1]^2 + 4.5*newx[2]^2)
```

[152]:

$$0.5\left(-1.0\alpha x_1 + x_1\right)^2 + 364.5\left(-\alpha x_2 + 0.1111111111111111x_2\right)^2$$

[153]:
```
Jnew=factor(Jnew)
```

[153]:

$$364.5\left(0.00137174211248285\alpha^2x_1^2 + 1.0\alpha^2x_2^2 - 0.00274348422496571\alpha x_1^2 - 0.222222222222222\alpha x_2^2 + 0.00137174211$$

[154]:
```
dJda=(diff(Jnew,alpha))
```

[154]:

$$1.0\alpha x_1^2 + 729.0\alpha x_2^2 - 1.0x_1^2 - 81.0x_2^2$$

substitute with [5,1]

[155]:
```
dJda(x1=>5, x2=>1)
```

[155]:

$$754.0\alpha - 106.0$$

[ ]:
```

```

[156]:
```
alpha_star=106/754
```

[156]: 0.14058355437665782

4 On the Newton method

For a given starting point x(0) = [x1, x2] , compute the Newton direction at this point.

[159]:
```
function gradient2(J,x1,x2)

dJdx1=diff(J,x1)  ;

dJdx2=diff(J,x2) ;
dJ=[dJdx1; dJdx2]
return dJ
end
```

[159]: gradient2 (generic function with 1 method)

[160]:
```
function hessian2(dJ,x1,x2)

Hdx11=diff(dJ[1],x1);
Hdx12=diff(dJ[1],x2);

Hdx21=diff(dJ[2],x1);

Hdx22=diff(dJ[2],x2);
H=[Hdx11 Hdx12; Hdx21 Hdx22]
return H
end
```

[160]: hessian2 (generic function with 1 method)

[161]:
```
J=0.5*x1^2 + 4.5*x2^2
```

[161]:

$$0.5x_1^2 + 4.5x_2^2$$

[162]:
```
dJ=gradient2(J,x1,x2)
```

[162]:

$$\begin{bmatrix} 1.0x_1 \\ 9.0x_2 \end{bmatrix}$$

[163]:
```
H=hessian2(dJ,x1,x2)
```

[163]:

$$\begin{bmatrix} 1.0 & 0 \\ 0 & 9.0 \end{bmatrix}$$

give newton's direction

[164]:
```
d_0=-inv(H)*dJ
```

[164]:

$$\begin{bmatrix} -1.0x_1 \\ -1.0x_2 \end{bmatrix}$$

it seems always negative? right ? is a descent direction for J at the point x(0)

[165]:
```
xnew=x+d_0
```

[165]:

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Hence, one gets in one iteration the optimal solution independently of the starting point (this is the case for all quadratic minimization problems)

5 Same question with a new function

For a given starting point x(0) = [x1, x2] , compute the Newton direction at this point.

[166]:
```
J=x1^4 + 2*x2^2
```

[166]:

$$x_1^4 + 2x_2^2$$

[167]:
```
dJ=gradient2(J,x1,x2)
```

[167]:

$$\begin{bmatrix} 4x_1^3 \\ 4x_2 \end{bmatrix}$$

[168]:
```
H=hessian2(dJ,x1,x2)
```

[168]:

$$\begin{bmatrix} 12x_1^2 & 0 \\ 0 & 4 \end{bmatrix}$$

[169]:
```
d_0=-inv(H)*dJ
```

[169]:

$$\begin{bmatrix} -\frac{x_1}{3} \\ -x_2 \end{bmatrix}$$

it seems always negative? right ? is a descent direction for J at the point x(0)

[170]:
```
xnew=x+d_0
```

[170]:

$$\begin{bmatrix} \frac{2x_1}{3} \\ 0 \end{bmatrix}$$

One iteration of the Newton method is not enough to converge to the optimal solution.

[ ]:
```
```