Project

1 _ Visualization

```
clear ; clc ; close all ;

h = 0.02 ;
x = 0:h:5;
y = 0:h:5 ;
[X, Y] = meshgrid (x,y);

U = (X-2).^2 + (Y-3).^2;

G = exp (-U)

F = G + 0,5*cos(X).*sin(Y);
```

```
figure;
surf(X, Y, F);
shading interp;
xlabel('x'); ylabel('y'); zlabel('f(x,y)')
title('surface')

figure;
contourf(X,Y,F,30);
colorbar
xlabel('x'); ylabel('y')
title('courbe niveau').
```

2 - Analytical Calculations

- $\nabla f(x,y) = (0;0)$

$\Longleftrightarrow (2,3)$

- $Hf(2;3)$

$$= \begin{pmatrix} -2-0{,}5\cos(2)\sin(3) & -0{,}5\sin(2)\cos(3) \\ -0{,}5\sin(2)\cos(3) & -2-0{,}5\cos(2)\sin(3) \end{pmatrix}$$

$$D = (-2-0{,}5\cos(2)\sin(3))^2 - (-0{,}5\sin(2)\cos(3))^2$$

$$= 4 + \cos(2)\sin(3) + 0{,}25\cos^2(2)\sin^2(3)$$

$$D = \left(-2 - 0{,}5\cos(2)\sin(3)\right)^2 - \left(-0{,}5\sin(2)\cos(3)\right)^2$$

$$= 4 + \cos(2)\sin(3) + 0{,}25\cos^2(2)\sin^2(3) - 0{,}25\sin^2(2)\cos^2(3)$$

$$= 4 + \cos(2)\sin(3) + 0{,}25\left(\cos^2(2)\sin^2(3) - \sin^2(2)\cos^2(3)\right)$$

$$= 4 + \cos(2)\sin(3) + 0{,}25\,x$$

$$\left[\left(\cos(2)\sin(3) + \sin(2)\cos(3)\right)\left(\cos(2)\sin(3) - \sin(2)\cos(3)\right)\right]$$

$$= 4 + \cos(2)\sin(3) + 0{,}25\sin(5)\sin(1)$$

$\simeq 3,54 > 0$

et $f_{xx} \simeq -1,975 < 0$

donc $\underbrace{\text{local maximum.}}_{\text{near } (2;3).}$

③ – Numerical

①

- $-2(x-2)e^{-[(x-2)^2+(y-3)^2]} - 0,5\sin(x)\sin(y) = 0$

$$\iff x-2 = \frac{0,5\sin(x)\sin(y)}{-2e^{-[(x-2)^2+(y-3)^2]}}$$

$$\Leftrightarrow x = \underbrace{2 - \frac{1}{4}\sin(x)\sin(y)\, e^{(x-2)^2+(y-3)^2}}_{g_1(x,y)}$$

$$\bullet\ -2(y-3)\, e^{-[(x-2)^2+(y-3)^2]} + 0,5\cos(x)\cos(y) = 0$$

$$\Leftrightarrow y-3 = \frac{-0,5\cos(x)\cos(y)}{-2\, e^{-[(x-2)^2+(y-3)^2]}}$$

$$\Leftrightarrow y = \underbrace{3 + \frac{1}{4}\cos(x)\cos(y)\, e^{(x-2)^2+(y-3)^2}}_{g_2(x,y).}$$

on pose :

$$g_1(x,y) = 2 - \frac{1}{4}\sin(x)\sin(y)\, e^{(x-2)^2+(y-3)^2}$$

et 2

$$g_2(x,y) = 3 + \frac{1}{4}\cos(x)\cos(y)\, e^{(x-2)^2+(y-3)^2}$$

avec $(x_0, y_0) = (2 ; 3)$ 3

1
```
function val = f_xy(x,y)
    G = exp(-((x-2)^2 + (y-3)^2));
    val = G + 0,5*cos(x)*sin(y);
end
```

②
```
function g = grad_f(x,y)
    G = exp(-((x-2)^2 + (y-3)^2));
    fx = -2*(x-2)*G - 0,5*sin(x)
         *sin(y);

    fy = -2*(y-3)*G - 0,5*cos(x)
         * cos(y);

    g = [fx ; fy];
end
```

③
```
function H = hess_f(x,y)
    G = exp(-((x-2)^2 + (y-3)^2));
    fxx = (4*(x-2)^2 - 2)*G
          - 0,5*cos(x)*sin(y);
```

```
fyy = (4*(y-3)^2-2)*G - 0,5*
                  cos(x)*sin(y);

fxy = 4*(x-2)*(y-3)*G - 0,5*
                  sin(x)*cos(y);

H = [fxx, fxy; fxy, fyy]

end
```

Fixed Point Iteration

```
clear; clc; close all;

function [x,y,iter,hist]
   = fixed_point(x0,y0,tol,maxIter

   x = x0;    y = y0;
   hist = zeros(maxIter,4);
```

```
for iter = 1 : maxIter
    G = exp(((x-2)^2 + (y-3)^2));
    g_1 = 2 - (sin(x) * sin(y) * G)/4;
    g_2 = 3 + (cos(x) * cos(y) * G)/4

    stepNorm = norm([g_1 - x; g_2 - y]);

    gradNorm = norm(grad_f(g_1, g_2));

    hist(iter, :) = [g_1, g_2, stepNorm, gradNorm];

    x = g_1 ; y = g_2 ;
```

```
if stepNorm < tol || gradNorm < tol
        hist = hist(1:iter,:);
        return;

    end

  end

end
```

Fixed Point Relaxed

```
function [x,y, iter, hist] = fixed_point(x0, y0, a, tol, maxIter)

    x = x0;      y = y0;
    hist = zeros(maxIter, 4);

    for iter = 1 : maxIter
```

```
g = grad_f(x,y);
x_new = x - a*g(1);
y_new = y - a*g(2);
stepNorm = norm([x_new - x; y_new
 - y]);
gradNorm = norm(grad_f(x_new, y_new));
hist(iter,:) = [x_new, y_new, stepNorm,
 gradNorm];

x = x_new; y = y_new;

if stepNorm < tol || gradNorm
 < tol
    hist = hist(1:iter,:);
    return;
end end
end end
```

Newton

```
function [x, y, iter, hist]
  = newton (x0, y0, tol, maxIter).

x = x0;      y = y0;
hist = zeros(maxIter, 6);

for iter = 1 : maxIter

    g = grad_f(x,y);
    H = hess_f(x,y);
    detH = det(H);

if  abs (detH) < 1e-14

    warning ('Hessienne non inversible')

    hist = hist(1:iter-1, :);
    return
end.
```

```
step = H \ g;

x_new = x_step(1);

y_new = y_step(2);

stepNorm = norm([x_new - x; y_new
                                  - y]);

gradNorm = norm(grad_f(x_new, y_new));

hist(iter, :) = [x_new, y_new, stepNorm,
                              gradNorm,
                    detH, f_xy(x_new, y_new)];

x = x_new; y = y_new;

  if stepNorm < tol || gradNorm
                                < tol
      hist = hist(1:iter, :);
      return;
end end end
end
```