TP 6

Fonctions de plusieurs variables

Compétences attendues.

✓ Tracer le graphe d'une fonction de plusieurs variables.

✓ Tracer un champ de gradients et des lignes de niveau d'une fonction de plusieurs variables.

Mathieu Mansuy - Professeur de Mathématiques en ECS2 au Lycée Louis Pergaud (Besançon)
mansuy.mathieu@hotmail.fr

1 Représentation graphique

1.1 Saisie de f

Dans toute la suite, f désigne une fonction de $\mathbb{R}^2$ dans $\mathbb{R}$. Pour définir la fonction f sur `Scilab`, on utilise la code suivant dans l'éditeur de texte `Scinote` :

```
function z = f(x,y)
    z = ...
endfunction
```

Exercice 1
Saisir sur `Scilab` la fonction $f : (x, y) \in \mathbb{R}^2 \mapsto \sin(x)\sin(y)$.

On utilise le code :

```
function z = f(x,y)
    z= sin(x)*sin(y)
endfunction
```

1.2 Graphe de f

Définition.

Soient `x`, `y` des vecteurs de tailles respectives n et m, `z` une matrice de taille $n \times m$, et `f` une fonction prenant deux arguments réels.

- `plot3d` trace une surface brisée en reliant les points de l'espace de coordonnées $(x_i, y_j, z_{i,j})$ pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$.
- `fplot3d` trace une surface brisée en reliant les points de l'espace de coordonnées $(x_i, y_j, f(x_i, y_j))$ pour tout $1 \leq i \leq n$ et $1 \leq j \leq m$.

Méthode.

Pour tracer la représentation graphique de f sur $[a, b] \times [c, d]$, on procèdera comme suit :

- *On crée deux vecteurs* `x` *et* `y` *découpant les intervalles $[a, b]$ et $[c, d]$ en n petits intervalles de même longueur comme suit :*

  ```
  x=linspace(a,b,n) ; y=linspace(c,d,n)
  ```

  On obtient ainsi un maillage $((x_i, y_j))_{1\leq i,j\leq n}$ du domaine $[a, b] \times [c, d]$.
- *On obtient alors la représentation graphique de f avec l'instruction* `fplot3d(x,y,f)`*.*

Remarque. Nous utiliserons en fait `fplot3d1` au lieu de `fplot3d`, qui fournit des graphiques plus colorés. Toutefois, c'est la seconde qui est au programme (mais leurs fonctionnements sont identiques).

Remarque. On peut éventuellement se passer de la fonction `fplot3d` en procédant de la manière suivante :

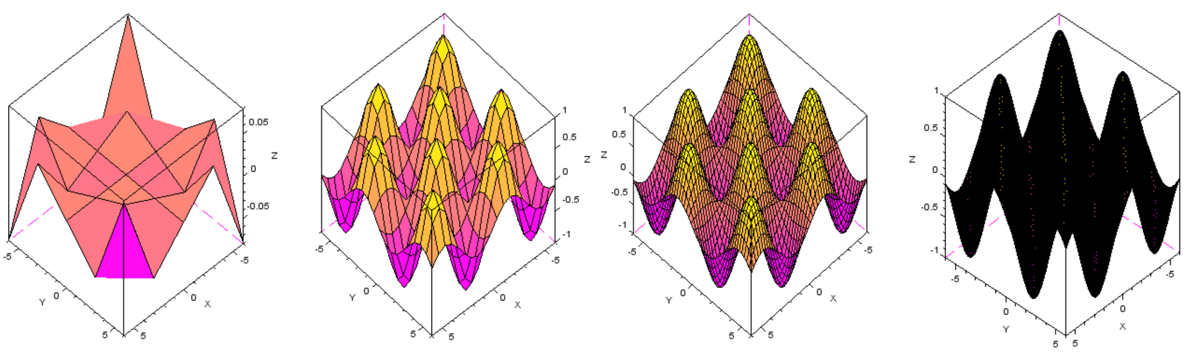
- on définit une matrice `z` contenant les images de (x, y) par f (taille $n \times m$) :

```
z = feval(x,y,f)
```

- on obtient alors la représentation graphique de f avec l'instruction `plot3d(x,y,z)`.

Exercice 2

Représenter le graphe de la fonction $f(x, y) = \sin(x)\sin(y)$ sur $[-6, 6] \times [-6, 6]$. On pourra tester successivement $n = 5, 20, 50, 200$ pour définir les vecteurs `x` et `y`.

> On utilise les lignes de commandes suivantes :
>
> ```
> n = [5,20,50,200]
> for i = 1:4
>     x=linspace(-6,6,n(i))
>     y = x
>     subplot(1,4,i)
>     fplot3d1(x,y,f)
> end
> ```
>
> La boucle `for` n'est pas indispensable, et on peut changer manuellement la valeur de `n` et exécuter de nouveau à chaque fois. La commande `subplot` permet de tracer les graphes les uns à côté des autres. On obtient :
>
> 
>
> Le graphe de droite nous apparait en noir du fait de son maillage très petit ($n = 200$).

Remarques.

- Dans la suite, on prendra $n = 50$ qui est un bon compromis entre la qualité de la représentation du graphe de f et la complexité de calcul pour `Scilab`.
- On peut faire tourner le graphe afin de l'orienter à sa guise à l'aide du clic droit.
- On peut changer de couleurs en ajoutant à la fin du script les commandes suivantes :

  ```
  g = gcf() ;
  g.color_map = rainbowcolormap(128) ;
  ```

  Si les couleurs ne sont toujours pas à vôtre goût, vous pouvez remplacer `rainbowcolormap` par `hotcolormap, coolcolormap, jetcolormap, summercolormap, springcolormap,` ...

1.3 Lignes de niveau

Définition.

Soient `x`, `y` des vecteurs et `f` une fonction prenant deux arguments réels.
Pour $n \in \mathbb{N}^*$, la commande `contour(x,y,f,n)` trace n lignes de niveau de la fonction f.

Exercice 3

On considère la fonction $f(x, y) = (x^3 + y^2)e^{-(x^2+y^2)}$. Tracer le graphe de f ainsi que n lignes de niveau pour $n = 5$ puis 10, sur le domaine $[-3, 3] \times [-3, 3]$.

Pour tracer les lignes de niveau dans une autre fenêtre que le graphe de f, on pourra commencer par utiliser `fplot3d1`*, puis ouvrir un autre graphique à l'aide de la commande* `scf()` *et utiliser* `contour`*.*

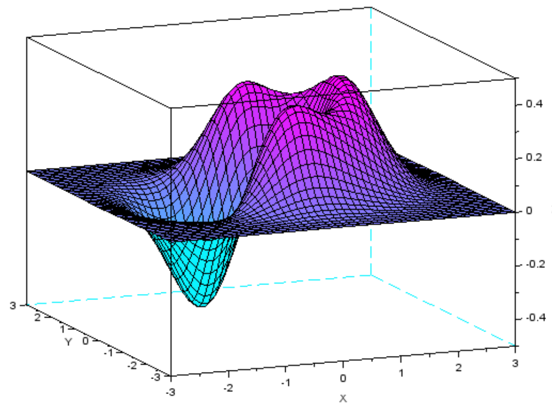
On utilise le code suivant :

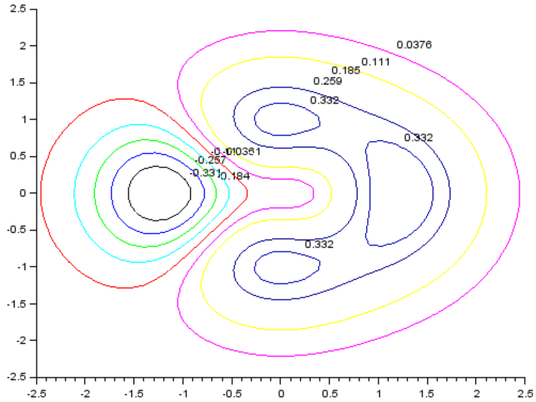
```
//Définition de la fonction
function z=f(x,y)
    z= (x^3+y^2)*exp(-x^2-y^2)
endfunction

//Graphe de la fonction
x = linspace(-3,3,50)
y = x
fplot3d1(x,y,f)
h = gcf();
h.color_map = coolcolormap(64)

//Lignes de niveau
scf()
contour(x,y,f,10)
```

On obtient :





2 Calcul différentiel du premier ordre

2.1 Dérivées partielles, gradient

Exercice 4

On considère la fonction $f(x,y) = (x^3+y^2)e^{-(x^2+y^2)}$.

1. Justifier que f est de classe $\mathscr{C}^1$ sur $\mathbb{R}^n$, puis déterminer ses dérivées partielles $\partial_1 f$ et $\partial_2 f$.

2. Définir en `Scilab` les fonctions `df1` et `df2` qui donnent les dérivées partielles $\partial_1 f$ et $\partial_2 f$ respectivement.

1. $(x,y) \mapsto x^3+y^2$ et $(x,y) \mapsto -x^2-y^2$ sont polynomiales donc $\mathscr{C}^1$, et $u \mapsto e^u$ est $\mathscr{C}^1$ sur $\mathbb{R}$. Par produit et compositions de fonctions $\mathscr{C}^1$, f est $\mathscr{C}^1$ sur $\mathbb{R}^2$. De plus on a pour tout $(x,y) \in \mathbb{R}^2$:

$$\partial_1 f(x,y) = (3x^2 - 2x(x^3+y^2))e^{-x^2-y^2} \quad \text{et} \quad \partial_2 f(x,y) = (2y - 2y(x^3+y^2))e^{-x^2-y^2}.$$

2. On utilise le code suivant :

```
function z = d1f(x,y)
    z = (3*x^2-2*x*(x^3+y^2))*exp(-x^2-y^2)
endfunction

function z = d2f(x,y)
    z = (2*y-2*y*(x^3+y^2))*exp(-x^2-y^2)
endfunction
```

On souhaite représenter le *champ de gradients* associé à une fonction $f : \mathbb{R}^2 \to \mathbb{R}$, c'est à dire :

- on fixe un réseau de points $(x_i, y_j)_{(i,j)[\![1,n]\!]\times[\![1,m]\!]}$ du plan ;
- en chaque point (x_i, y_j) du réseau, on représente le vecteur gradient $\nabla f(x_i, y_j)$.

On utilise pour cela la commande `champ`.

Définition.

Soient `x`, `y` des vecteurs de tailles respectives n et m, `vx`, `vy` des matrices de tailles $n \times m$. La commande `champ(x,y,vx,vy)` trace en chaque point `(x(i),y(j))` du plan le vecteur de coordonnées `(vx(i,j),vy(i,j))` pour tout $(i,j) \in [\![1,n]\!] \times [\![1,m]\!]$.

Méthode.

Pour tracer le champ de gradient associé à une fonction $f : [a,b] \times [c,d]$, on procèdera ainsi :

- *On crée **x** et **y** des vecteurs de taille $n \times m$ qui définissent un réseau (x_i, y_j) de $[a,b] \times [c,d]$.*
- *On définit ensuite **D1** la matrice contenant les images des (x_i, y_j) par $\partial_1 f$ et **D2** la matrice contenant les images des (x_i, y_j) par $\partial_2 f$.*

  On pourra utiliser pour cela `feval(x,y,d1f)` *et* `feval(x,y,d2f)`.
- *On obtient alors le champ de vecteurs de f avec l'instruction* `champ(x,y,D1,D2)`.

Exercice 5

Tracer sur un même graphique le champ de gradients et les lignes de niveau de $f(x,y) = (x^3+y^2)e^{-(x^2+y^2)}$. On utilisera pour cela le réseau défini par $x_i = -3+i$ et $y_j = -3+j$ pour $0 \leq i,j \leq 6$.

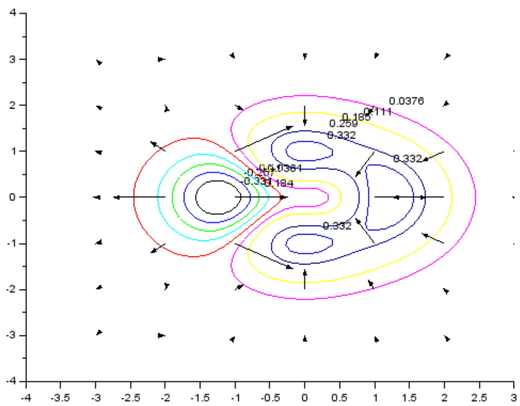
Interpréter le résultat : direction et norme du gradient, repérer le(s) point(s) critique(s) éventuel(s) et leur nature, positions relatives du gradient et des lignes de niveau. En particulier, quel résultat évoqué en cours retrouvez vous ?

On utilise le code suivant :

```
\\Lignes de niveau
x=linspace(-3,3,50)
y=x
contour(x,y,f,10)

\\Champ de gradients
u=-3:3
v=-3:3
D1 = feval(u,v,d1f)
D2 = feval(u,v,d2f)
champ(u,v,D1,D2)
```

On obtient :



On notera que le gradient en un point indique la direction de plus fort accroissement de f en ce point. L'absence de gradient indique un point où le gradient est nul, c'est-à-dire un point critique. Enfin, on remarque que le gradient est orthogonal aux (tangentes aux) lignes de niveau.

2.2 Plan tangent

Méthode.

Pour représenter le plan affine tangent au graphe de f en (x_0, y_0), on définit la fonction affine :

$$t_{(x_0,y_0)} : (x, y) \mapsto f(x_0, y_0) + \partial_1(f)(x_0, y_0) \times (x - x_0) + \partial_2(f)(x_0, y_0) \times (y - y_0)$$

On trace sa représentation graphique sur le même graphe que f (au moins au voisinage de (x_0, y_0)).

Exercice 6

On considère la fonction $f(x, y) = (x^3+y^2)e^{-(x^2+y^2)}$ sur $[-3, 3]\times[-3, 3]$. Tracer la représentation graphique de f ainsi que celle des plans affines tangents aux points $(0, 0)$, $(0, 1)$ puis $(1, 2)$.

Après calcul des équations des plans tangents, on peut entrer le code suivant :

```
//Plan tangent en (0,0)
function z = plan1(x,y)
    z = 0
endfunction

u = linspace(-1,1,10)
v = linspace(-1,1,10)
fplot3d1(u,v,plan1)

//Plan tangent en (0,1)
function z = plan2(x,y)
    z = exp(-1)
endfunction

u = linspace(-1,1,10)
v = linspace(0,2,10)
fplot3d1(u,v,plan2)

//Plan tangent en (1,2)
function z = plan3(x,y)
    z = exp(-5)*(5-7*(x-1)-16*(y-2))
endfunction

u = linspace(0.5,1.5,10)
v = linspace(1.5,2.5,10)
fplot3d1(u,v,plan3)
```

On obtient :