Cours 15 : 10 avril 2019 Lignes de niveau

Le but est de donner une méthode pour tracer les lignes de niveau d'une fonction à deux variables par exemple

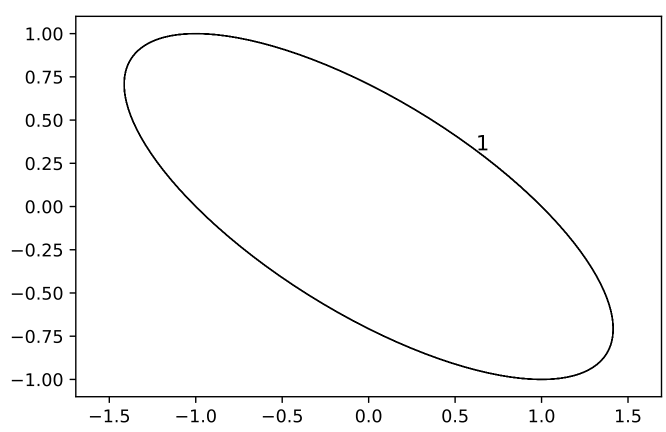
$$f(x,y) = x^2 + 2\,y^2 + 2\,x\,y$$

1. Préparation des fonctions élémentaires : avec $f(x,y) = x^2 + 2\,y^2 + 2\,x\,y$:
   Attention : par la suite il suffira de changer $\boldsymbol{f}$ en $\boldsymbol{g}$ et $\boldsymbol{\nabla f}$ en $\boldsymbol{\nabla g}$ pour avoir les lignes de niveau d'une autre fonction $\boldsymbol{g}$
   (a) Enregistrer la fonction `def f(u) :`
   qui prend pour seule entrée la liste ou tableau $u = [x, y]$ et retourne $f(x, y)$
   (b) Enregistrer `def Nabla(u) :`
   qui prend pour seule entrée la liste pou tableau $u = [x, y]$ et retourne le vecteur $\nabla f(x, y)$ sous forme de tableau de type array de numpy
   (c) Enregistrer `def Vunit(u) :`
   qui prend pour seule entrée la liste ou tableau $\vec{u} = [x, y]$ et retourne le vecteur unitaire $\vec{\omega} = \dfrac{\vec{u}}{||\vec{u}||}$ sous forme de tableau de type array de numpy

2. Construire une fonction `def points(n,pas,x,y) :`
   qui renvoie une liste $A = [A_i]_{i\in[[0,n-1]]}$ contenant n points A_i sur la ligne de niveau $\mathcal{C}_p$ où $k = f(A_0)$ en commençant par $A_0 = [x, y]$ et où les points suivants A_i seront construits par récurrence de la manière suivante :
   (a) En A_i supposé sur la ligne de niveau $\mathcal{C}_0$ on calcule vecteur tangent unitaire $\overrightarrow{T}$ à partir de $\nabla f(A_i)$
   (b) On calcule le point intermédiaire B_i tel que $\overrightarrow{A_iB_i} =$ `pas` $\overrightarrow{T}$
   (c) On calcule A_{i+1} tel que $\boldsymbol{\overrightarrow{OA_{i+1}} = \overrightarrow{OB_i} + \lambda\nabla f(B_i)}$
   Où λ est tel que, d'après la formule de Taylor-Young à l'ordre 1 , le point A_{i+1} soit sur la ligne de niveau $\mathcal{C}_k$ où $k = f(A_0)$. La valeur de λ est supposée être faible devant celle du `pas`

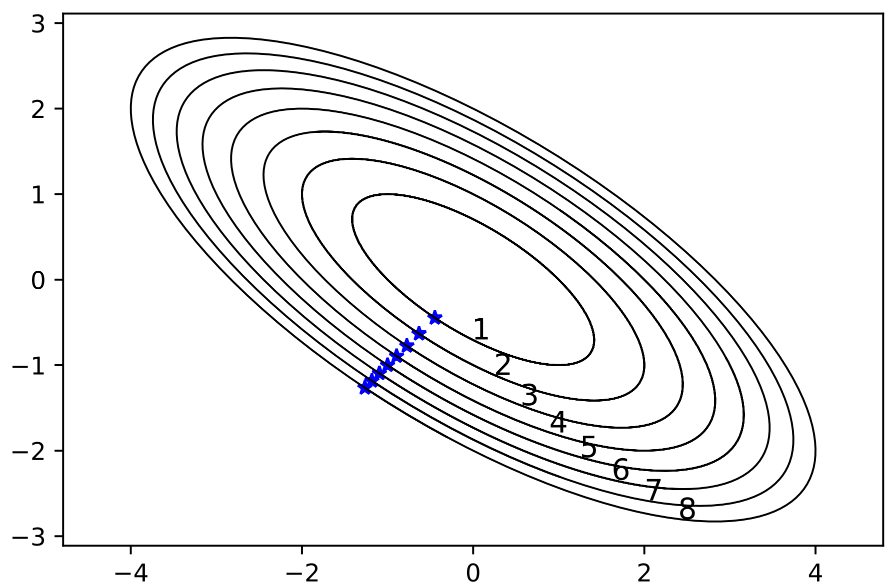
   Par exemple pour 10 points avec un pas de 0.5 à partir de (1,0) on trouve
   >>> A=points(10,0.5,1,0)

   >>> A=[[1. , 0.], [0.62, 0.32], [0.20, 0.59], [-0.23, 0.81], [-0.70, 0.96], [-1.18, 0.98], [-1.42, 0.84], [-1.44, 0.52], [-1.15, 0.14], [-0.79 , -0.18]]



3. À partir de `A= points(n,pas,1,0)` avec `n=3000 :` et `pas=1/100 :` , tracer la ligne de niveau $\mathcal{C}_1$ ci contre qui passe par le point $A_0 = [1, 0]$

4. Donner la méthode de dichotomie `def dicho(g,a,b):` qui retourne à la précision 10^{-6} près une solution de $g(x) = 0$ sur un intervalle $[a, b]$ si $g(a)g(b) \leq 0$ sinon elle ne retourne rien
5. Tester votre fonction en donnant après reconnaissance graphique les 3 solutions de l'équation $e^x = 3x^2$ à 10^{-6} près .
   Vérifier que le produit des solutions vaut : `-1.559156`
6. Améliorer la fonction `def dicho(g,a,b):` pour qu'elle retourne une solution si elle existe de $g(x) = 0$ à 10^{-6} près en cherchera d'abord un intervalle $[\boldsymbol{c}, \boldsymbol{d}] \subset [\boldsymbol{a}, \boldsymbol{b}]$ **tel que** $\boldsymbol{d} - \boldsymbol{c} \leq \dfrac{\boldsymbol{b} - \boldsymbol{a}}{\mathbf{100}}$ **et** $\boldsymbol{g(c)g(d)} \leq \mathbf{0}$**.**
   Si aucun changement de signe $\boldsymbol{g(c)g(d)} \leq \mathbf{0}$ de ce type n'est repéré , la fonction renvoie à nouveau l'ensemble vide.
7. Ecrire une procédure `def lignes(n,k0,k1):` qui dessine les lignes de niveau de $f(x, y) = x^2 + 2y^2 + 2xy = k$ pour k allant de $k0$ à $k1$ en recherchant les départs de la ligne de niveau sur la droite $y = x$ du plan .



8. Autre exemple : Dessiner les lignes de niveau de $h(x, y) = x^4 + y^4 - 4xy$