M1 MINT et M1 MMM
Semestre 7, Année 2015
Mr DUMAS

CC2 OPTIMISATION

Exercice 1 –

On s'intéresse ici à un nouveau principe de recherche linéaire pour une méthode à direction de descente.

Soit f une fonction C^1 de $\mathbb{R}^n$ dans $\mathbb{R}$, telle que $\lim\limits_{||x||\to+\infty} f(x) = +\infty$. On note g le gradient de la fonction f défini de $\mathbb{R}^n$ dans $\mathbb{R}^n$. On suppose que g est Lipschitz continu sur $\mathbb{R}^n$ avec une constante de Lipschitz égale à L.

On rappelle qu'une méthode de descente consiste à définir une suite $(x_k)_{k\in\mathbb{N}}$ partant d'un point $x_0 \in \mathbb{R}^n$ avec la relation:

$$x_{k+1} = x_k + t_k d_k$$

où d_k est une direction de descente (telle que $< d_k, g(x_k) >< 0$) et t_k est le pas correspondant.

Ici, le pas est donné par la règle suivante: $t_k = 0$ si $g(x_k) = 0$ et sinon:

$$t_k = \inf\{t \geq 0, \quad h_k'(t) = 0,\ h_k(t) < h_k(0)\}$$

où pour tout $t \geq 0$, $h_k(t) = f(x_k + td_k)$.

1. Prouver que la condition choisie est valide, c'est à dire que t_k existe pour tout $k \geq 0$. Représenter graphiquement un exemple de tel pas pour une fonction arbitraire (non convexe).

2. On dit que la condition Z est vérifié pour un pas de descente si on a:

$$f(x_{k+1}) \leq f(x_k) - C||g(x_k)||^2 \cos^2(\theta_k)$$

pour une constante $C > 0$ donnée indépendante de k et où

$$\cos(\theta_k) = \frac{-< d_k, g(x_k) >}{||g(x_k)||.||d_k||}$$

On cherche à prouver que la condition Z est bien vérifiée pour le pas choisi ici.

2 a) En utilisant une égalité de Taylor, prouver que pour tout $t \in [0, t_k]$

$$h(t_k) \leq h(t) \leq h(0) + t(d_k, g(x_k)) + t^2 L\frac{||d_k||^2}{2} \tag{1}$$

2b) Trouver la valeur t, nommée t_{min} où le terme à droite dans l'inégalité (1) est minimal.

2c) Prouver que

$$0 \leq (d_k, g(x_k)) + t_k L||d_k||^2$$

et en déduire que

$$f(x_{k+1}) \leq f(x_k) - \frac{(d_k, g(x_k))^2}{2L||d_k||^2}$$

2c) Conclure.

3. Prouver que la condition Z implique:

$$\sum_{k=1}^{+\infty} ||g(x_k)||^2 \cos^2(\theta_k) < +\infty$$

4. On suppose que $d_k = -g(x_k)$ (direction de descente du gradient). Montrer que la méthode de descente ainsi construite converge en un sens à préciser.

Exercice 2 – A la manière des algorithmes génétiques, la méthode DE recherche de manière stochastique le minimum global d'une fonction $J : \mathbb{R}^n \to \mathbb{R}$.

DE fait évoluer une population de N_{pop} elements (ou individus) avec l'algorithme suivant (où $CR \in [0, 1]$ et $F \in [0, 2]$ sont deux paramètres):

(i) Initialisation aléatoire de N_{pop} elements

(ii) De la génération 1 à la generation N_{gen}:

(iii) Pour chaque individu $x \in \mathbb{R}^n$:

- Choisir alétoirement trois éléments a, b et c dans la population, distincts entre eux et distincts de x.
- Tirer i_0 indice aléatoire dans $\{1, ..., n\}$ et calculer $y = (y_1, ..., y_n)$ comme suit:

  $$\forall i \in \{1, ..., n\}, \quad y_i = a_i + F(b_i - c_i) \text{ si } (r_i < CR) \text{ ou } (i = i_0), \text{ sinon } y_i = x_i$$

  où r_i est choisi alétoirement dans $[0, 1]$.
- Si $J(y) < J(x)$, remplacer x par y dans la population.

(iv) Fin d'une génération

1. Quels sont les principaux points communs et les quelles sont les principales différences de l'algorithme DE par rapport à un algorithme génétique?

2. Interpréter les paramètres CR et F pour l'algorithme. Quelles valeurs extrêmes peuvent-ils prendre?

3. Le script suivant propose une implémentation de l'algorithme DE en Scilab:

```
function y=f(x)
    y=n+sum(x.^2-cos(2*%pi*x));
endfunction
////////parametres////////////////
Npop=40;
Ngen=20;
n=2;
CR=0.2//
F=0.8//
////////////////////////////////////
//
A=zeros(Npop,n+1); // matrice de population
//
A=10*rand(Npop,n+1)-5*ones(Npop,n+1);
//
//evaluation
y=[];
for j=1:Npop
    y(j)=f(A(j,1:n));
end
A(:,n+1)=y;
/////////////////////////////////////////////
for i=1:Ngen
//
[u,v]=gsort(A(:,n+1));
A=A(v,:); // rangement du plus mauvais au meilleur
<<<<     disp('meilleur element'),disp(     )
<<<<<   disp('meilleure valeur'),disp(A(    )
//
  for k=1:Npop
    i1=1;i2=1;i3=1;
    while (i1==i2)| (i1==i3)|(i2==i3)| (i1==k)| (i2==k)| (i3==k)
     i1=int(Npop*rand())+1;
     i2=int(Npop*rand())+1;
     i3=int(Npop*rand())+1;
```

```
    end
<<<<<<<    a=    ;b=      ;c=     ;x=      ;
<<<<<<     j0=
    y=[];
    for j=1:n
     if (rand()<CR)| (j==j0)
<<<<<      y(j)=
     else
<<<<<      y(j)=
     end
     val1=f(y);val2=f(x);
     if (val1<val2) then
       A(k,1:n)=y';A(k,n+1)=val2;
     end
    end
  end
end
```

Malheureusement, certaines lignes repérées par:

```
<<<<<<<
```

ont été effacées. Reconstituer les lignes correspondantes.

4. On souhaite tracer l'historique de décroissante de la meilleure valeur de f en fonction du nombre d'itérations. Rajouter les instructions manquantes pour cet affichage.