Travaux pratiques n°3
Méthodes de descente

Objectif : blabla

1 Forme générique des méthodes de descentes

Considérons le problème d'optimisation suivant :

$$\min_{x \in \mathcal{X}} J(x)$$

où $J : \mathbb{R}^n \to \mathbb{R}$ est la fonction objectif différentiable ou deux fois différentiable, convexe. Dans ce TP, on s'intéresse à une famille d'algorithmes d'optimisation se présentant sous la forme générique :

1. **Initialisation ($k = 0$) :** définir $x_0 \in \mathbb{R}^n$ et $B_0 \in \mathcal{M}_{n,n}(\mathbb{R})$.
2. **Itérations ($k \in \mathbb{N}$) :** tant que le critère d'arrêt n'est pas satisfait, calculer

$$\begin{cases} g_k & = \nabla J(x_k) \\ d_k & = -B_k\, g_k \\ x_{k+1} & = x_k + \tau_k\, d_k \end{cases}$$

   avec $B_k \in \mathcal{M}_{n,n}(\mathbb{R})$ et $\tau_k \in \mathbb{R}$.
3. **Terminaison ($k = k_{\max}$) :** si le critère d'arrêt est satisfait, renvoyer $x_{k_{\max}}$.

Pour pouvoir analyser dans de bonnes conditions le comportement des différentes méthodes, on peut être amené à garder en mémoire les suites générées $(x_k)_{0 \leq k \leq k_{\max}}$, $(g_k)_{0 \leq k \leq k_{\max}}$ et $(B_k)_{0 \leq k \leq k_{\max}}$. On va donc commencer par considérer le prototype de fonction suivant :

```
1 //fonction generique pour optimisation
```

Remarque : Il existe plusieurs manières de définir un critère d'arrêt. Dans ce TP, on en considérera deux :

1. nombre d'itérations maximal : on fixe le nombre d'itérations à l'avance ;
2. seuil sur le critère d'optimalité du premier ordre : on arrête les itérations lorsque $\|\nabla J(x_k)\| \leq \varepsilon$ avec $\varepsilon > 0$ un seuil fixé à l'avance.

Ce ne sont pas les critères les plus pertinents, mais ils ont l'intérêt d'être très simples à mettre en place.

Exercice 1 – Critère d'arrêt

Implémenter les deux critères d'arrêt mentionnés ; pour le premier, on fixera $k_{\max} = 20$ et pour le second $\varepsilon = 10^{-2}$. La fonction renvoie 1 si le critère d'arrêt est satisfait, et 0 sinon. Comment peut-on combiner ces deux critères pour arrêter les itérations dès que l'un des deux critères est satisfait ?

2 Débruitage de signaux 1D

Soit $v \in \mathbb{R}^n$. On considère le problème d'optimisation suivant :

$$\min_{x \in \mathbb{R}^n} J(x) = \frac{\lambda}{2} \|x - v\|_2^2 + \frac{1}{2} \|D(x)\|_2^2$$

où $D : \mathbb{R}^n \to \mathbb{R}^{n-1}$ est une application linéaire, définie pour tout $x = (x_{(i)})_{1 \leq i \leq n}$ par les différences finies

$$\forall\, i \in [\![1\,;\, n]\!], \qquad (D(x))_{(i)} = \begin{cases} x_{(i+1)} - x_{(i)} & \text{si } i \neq n \\ 0 & \text{sinon} \end{cases}$$

Ce problème apparaît lorsque l'on souhaite débruiter une courbe **lisse**. Le paramètre λ est un réel strictement positif, qui permet d'ajuster la régularité du signal débruité.

Exercice 2 – Théorie

(a) Montrer que J est différentiable et que

$$\forall\, x \in \mathbb{R}^n, \qquad \nabla J(x) = \lambda\,(x - v) + D^*(D(x))$$

(b) Vérifier que

$$\forall\, x \in \mathbb{R}^n, \qquad D(x) = \underbrace{\begin{pmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & -1 & 1 \\ 0 & \cdots & \cdots & 0 & 0 \end{pmatrix}}_{=D} \begin{pmatrix} x_{(1)} \\ \vdots \\ x_{(n)} \end{pmatrix}$$

(c) En déduire que

$$\forall\, x \in \mathbb{R}^n, \qquad \nabla J(x) = (\lambda\, I_n + D^\top D)\, x - v$$

(d) Montrer que J est deux fois différentiable et que

$$\forall\, x \in \mathbb{R}^n, \qquad \mathrm{Hess} J(x) = \lambda\, I_n + D^\top D$$

(e) Montrer que J est convexe et fortement convexe de module λ.

(f) En déduire que J admet un unique minimiseur x^*, donné par

$$x^* = (\lambda\, I_n + D^\top D)^{-1}\, v$$

On va maintenant générer des données v en ajoutant un bruit additif gaussien à un signal très lisse u. On peut choisir par exemple

$$\forall\, i \in [\![1\,;\, n]\!], \qquad u_{(i)} = -\exp(t_i) - 6\, t_i\, \sin t_i + 4\, t_i^2;$$

avec t_i des valeurs régulièrement espacées dans $\mathbb{R}$.

```
intervalle = [-1:0.025:1.5];
u = -exp(intervalle)-6*intervalle.*sin(intervalle)+4*intervalle.^2;
u = u';
v = u+0.1*rand(u,"normal");
save('v.dat',v); //sauvegarde dans le fichier v.dat
```

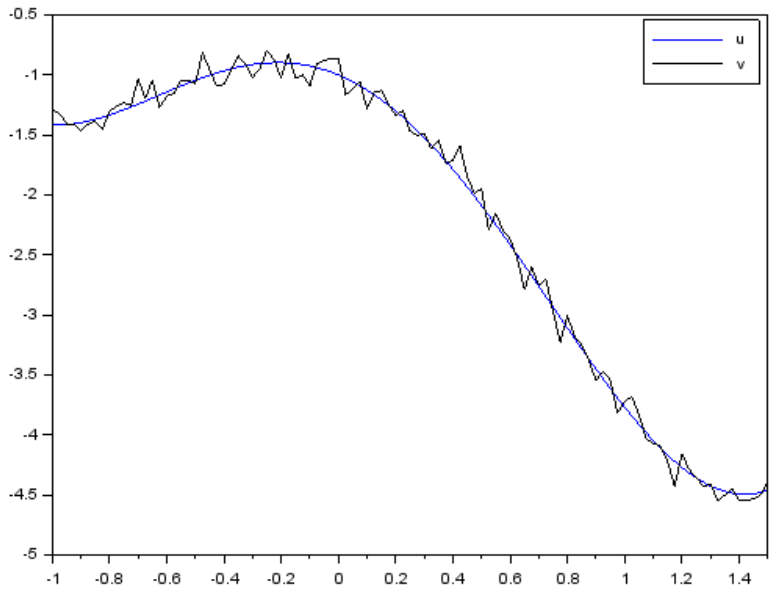

FIGURE 1 – Signal u et exemple de signal v.

Cette méthode pour le débruitage, qui consiste à introduire une fonctionnelle J à minimiser pour avoir une estimée du signal propre u, fait partie de la famille des *méthodes variationnelles*. Leur efficacité dépend principalement de deux éléments : l'adéquation de la fonctionnelle J au problème considéré (autrement dit, la proximité des minimiseurs de J avec l'objet recherché, ici u) et la capacité à calculer numériquement un minimiseur de J (en particulier, l'existence et l'applicabilité d'une méthode d'optimisation adaptée à J et son efficacité numérique). Les fonctionnelles J comprennent généralement deux types de termes. Tout d'abord un terme d'attache aux données (ou encore de fidélité). Ici, il s'agit du terme

$$\frac{\lambda}{2}\|x-v\|_2^2$$

qui traduit le fait que le signal reconstruit doit rester proche du signal observé. Plus λ est grand, et plus cette proximité sera grande. La proximité est ici définie à l'aide de la norme quadratique, car le bruit est (supposé) de nature gaussienne. Le second type de terme est celui des termes de régularisation, ici

$$\frac{1}{2}\|D(x)\|_2^2$$

Ils servent à encoder des propriétés connues de l'objet à estimer. Dans l'exemple considéré, le signal u initial est très lisse. Or, D n'est autre qu'une discrétisation de la dérivée. Autrement dit, le terme de régularisation choisi ici traduit le fait que la dérivée de u (vu comme la version continue du signal discrétisé) est localement faible. Lorsque le paramètre λ est faible, ce terme agit davantage que le terme d'attache aux données. Le signal reconstruit est alors d'autant plus lisse que λ est proche de zéro.

Exercice 3 – Optimum

Afficher sur un même graphique les vecteurs v, x^* et u. Tester avec différentes valeurs de λ (par exemple $\lambda \in \{1, 0.1, 0.001\}$).

3 Méthodes de gradient

Dans les méthodes de gradient, on choisit

$$\forall\, k \in \mathbb{N}, \qquad B_k = I_n$$

Quant au pas de temps τ_k, on a vu dans le TP précédent que différents choix sont possibles. On commence par considérer le cas du pas fixe et du pas optimal.

Si la fonction J est convexe et à gradient lipschitzien de constante de Lipschitz L, alors la convergence de la méthode du gradient est assurée dès que le pas de temps τ est choisi strictement inférieur à $2/L$.

Exercice 4 – Régularité de J

(a) Montrer que $$|||D||| = \sup_{x \neq 0} \frac{\|D(x)\|_2}{\|x\|_2} \leq 2.$$

On pourra majorer la quantité $\|D(x)\|_2^2$ à l'aide de l'inégalité $(a-b)^2 \leq 2\,(a^2+b^2)$ valable pour tout couple $(a, b) \in \mathbb{R}^2$.

(b) Montrer que ∇J est lipschitzien, de constante de Lipschitz

$$L = \lambda + 4$$

Exercice 5 – Méthode du gradient à pas fixe

Mettre en œuvre la méthode du méthode à pas fixe.

Dans le cas du pas optimal, on choisit pour tout $k \in \mathbb{N}$

$$\tau_k \in \arg\min_{\tau \in \mathbb{R}} J(x_k + \tau\, d_k)$$

Exercice 6 – Pas optimal

(a) Justifier que la fonction réelle

$$\tau \mapsto J(x + \tau\, d)$$

est une fonction convexe infinie à l'infini

(b) En déduire que son minimiseur τ^* existe et est unique, et est caractérisé par la condition nécessaire et suffisante d'optimalité du premier ordre, qui s'écrit dans ce cas

$$\langle \nabla J(x + \tau^*\, d), d \rangle = 0$$

Montrer que $$\tau^* = -\frac{\langle \nabla J(x), d \rangle}{\lambda\, \|d\|_2^2 + \|D\, d\|_2^2}$$

Exercice 7 – Méthode du gradient à pas optimal

Mettre en œuvre la méthode du méthode à pas optimal.

D'autres choix de pas de temps sont possibles. Ainsi, pour la méthode de Barzilai–Borwein, on choisit pour tout $k \in \mathbb{N}$

$$\tau_k = \frac{\langle \Delta g_k, \Delta x_k \rangle}{\|\Delta g_k\|^2}$$

avec $\Delta x_k = x_k - x_{k-1}$ et $\Delta g_k = \nabla J(x_k) - \nabla J(x_{k-1})$.

Remarque : Attention : on voit que, pour calculer τ_k, on doit avoir déjà calculé deux points x_k et x_{k-1}. En pratique, cela signifie qu'il faut non seulement initialiser l'algorithme avec un point x_0, mais également avec un point précédent x_{-1} différent de x_0. On pourra choisir par exemple $x_{-1} = -x_0$ si $x_0 \neq 0$.

Exercice 8 – Méthode de Barzilai–Borwein

Mettre en œuvre la méthode de Barzilai–Borwein.

4 Méthodes de Newton et de quasi-Newton

Dans la méthode de Newton pour la recherche d'un point critique, on choisit

$$\begin{cases} B_k = (\mathrm{Hess} J)^{-1}(x_k) \\ \tau_k = 1 \end{cases}$$

Exercice 9 – Méthode de Newton

Mettre en œuvre la méthode de Newton.

Dans certaines applications, la matrice hessienne (et/ou son inverse) est coûteuse à calculer. Aussi, on peut choisir de la remplacer pour une matrice approchée : on parle de méthodes de quasi-Newton. Plus précisément, on remplace B_k soit par une approximation de la hessienne, soit par l'inverse d'une approximation de la hessienne. Ainsi, la méthode DFP (pour Davidon–Fletcher–Powel) utilise

$$\begin{cases} B_k = B_{k-1} + \dfrac{\Delta x_k \, (\Delta x_k)^\top}{\langle \Delta x_k, \Delta g_k \rangle} - \dfrac{(B_{k-1} \, \Delta g_k) \, (B_{k-1} \, \Delta g_k)^\top}{\langle B_{k-1} \, \Delta g_k, \Delta g_k \rangle} \\ \tau_k \in \arg\min\limits_{\tau > 0} J\big(x_k - \tau \, B_k \, \nabla J(x_k)\big) \end{cases}$$

tandis que la méthode BFGS (pour Broyden–Fletcher–Goldfarb–Shanno) s'écrit

$$\begin{cases} B_k = \left(B_{k-1}^{-1} + \dfrac{\Delta g_k \, (\Delta g_k)^\top}{\langle \Delta x_k, \Delta g_k \rangle} - \dfrac{(B_{k-1}^{-1} \, \Delta x_k) \, (B_{k-1}^{-1} \, \Delta x_k)^\top}{\langle B_{k-1}^{-1} \, \Delta x_k, \Delta x_k \rangle} \right)^{-1} \\ \tau_k \in \arg\min\limits_{\tau > 0} J\big(x_k - \tau \, B_k \, \nabla J(x_k)\big) \end{cases}$$

Exercice 10 – Méthodes de quasi-Newton

Mettre en œuvre les méthodes DFP et BFGS.

Exercice 11 – Comparaison

Comparer les différentes méthodes implémentées. On affichera pour chaque méthode les courbes suivantes :

- $(J(x_k))_{0\leq k\leq k_{\max}}$
- $(\|\nabla J(x_k)\|_2)_{0\leq k\leq k_{\max}}$
- $(\|x_k - x^*\|_2)_{0\leq k\leq k_{\max}}$
- $(\|x_k - u\|_2)_{0\leq k\leq k_{\max}}$