Exercises : Gradient stochastique

Optimisation pour l'apprentissage automatique

M2 IASD Apprentissage, 2024-2025

Dauphine | PSL
UNIVERSITÉ PARIS

Exercice 1: Random reshuffling

On considère un problème de minimisation du risque empirique de la forme

$$\text{minimiser}_{\boldsymbol{w}\in\mathbb{R}^d} f(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\boldsymbol{w}), \tag{1}$$

où chaque f_i est de classe $\mathcal{C}^1$ et dépend uniquement du $i$ème point d'un jeu de données à n éléments. On suppose aussi que la fonction objectif f est convexe, et $\mathcal{C}^{1,1}$. On supposera enfin que le problème (1) possède une solution, et on notera $f^* = \min_{\boldsymbol{w}\in\mathbb{R}^d} f(\boldsymbol{w})$.

On supposera que n est trop large pour que le jeu de données puisse être utilisé en entier lors d'une itération d'un algorithme, et on considère donc l'algorithme du gradient stochastique

$$x_{k+1} = x_k - \alpha_k \nabla f_{i_k}(x_k), \tag{2}$$

avec $\alpha_k > 0$ et $i_k \in \{1,\dots,n\}$. Le but de l'exercice est d'étudier les variantes basées sur le principe de *random reshuffling* (dont le principe peut être traduit par *rebattre les cartes*). Dans ces variantes, les indices $\{i_k\}$ sont tirés selon une perturbation aléatoire de $\{1,\dots,n\}$ qui est modifiée toutes les n itérations. Ainsi, à l'itération 0, une permutation aléatoire de $\{1,\dots,n\}$ définit $\{i_0,\dots,i_{n-1}\}$, puis à l'itération n, une autre permutation aléatoire définit $\{i_n,\dots,i_{2n-1}\}$, et ainsi de suite.

a) Rappeler la définition d'une époque (*epoch*). Avec la stratégie du *random reshuffling*, que peut-on garantir sur les points du jeu de données qui ont été tirés au cours de la première époque ?

b) Soit un indice d'itération k correspondant à la première itération d'une époque (càd $k = \ell n$ avec $\ell \in \mathbb{N}$).

   i) Montrer que
   $$\mathbb{E}_{i_k}\left[\nabla f_{i_k}(x_k)\right] = \nabla f(x_k).$$

   ii) Les indices $i_k,\dots,i_{k+n-1}$ n'étant pas indépendants, justifier que la propriété de la questions b)i) n'est pas vérifiée pour les autres itérations de l'époque.

c) Même sans l'hypothèse de la question b)i), on peut montrer des résultats de convergence pour un bon choix de taille de pas. Pour cela, on regarde la suite des itérés moyens $\{\bar{x}_K\}_K$, avec

$$\bar{x}_K = \frac{1}{K+1}\sum_{k=0}^{K} x_k \quad \forall K \in \mathbb{N}.$$

On peut ainsi montrer qu'après nK itérations, on a

$$\mathbb{E}\left[f\left(\bar{x}_{nK}\right)\right] - f^* \leq \mathcal{O}\left(\frac{1}{\sqrt{nK}}\right),$$

Comparer ce taux avec celui de la descente de gradient pour le même problème.

d) On considère une variantes par fournées du gradient stochastique, pour laquelle les indices sont tirés suivant une approche par *random reshuffling*. Si la taille de fournées est de n, quel algorithme retrouve-t-on ?

e) Comme l'approche par *random reshuffling* ne tire pas le même indice au sein d'une même époque, on peut vouloir utiliser de l'information des itérations précédentes pour améliorer le pas. Quelle technique vue lors des séances proposeriez-vous d'utiliser pour cela ?

Exercice 2 : Perte de Huber

On considère un jeu de données $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, où $n \geq 1$, $\boldsymbol{x}_i \in \mathbb{R}^d$ avec $d \geq 1$ et $y_i \in \mathbb{R}$. On cherche un modèle linéaire qui prédise au mieux chaque y_i à partir du $\boldsymbol{x}_i$ correspondant. On définit donc une famille de modèles paramétrée par $\boldsymbol{w} \in \mathbb{R}^d$ comme suit :

$$\begin{array}{rccl} h_{\boldsymbol{w}}: & \mathbb{R}^d & \to & \mathbb{R} \\ & \boldsymbol{x} & \mapsto & \boldsymbol{x}^{\mathrm{T}}\boldsymbol{w} = \sum_{i=1}^d [\boldsymbol{x}]_i [\boldsymbol{w}]_i. \end{array}$$

Pour un modèle $h_{\boldsymbol{w}}$, on considèrera que ce modèle prédit parfaitement y_i à partir de $\boldsymbol{x}_i$ si on a $\ell\left(h_{\boldsymbol{w}}(\boldsymbol{x}_i) - y_i\right) = \ell\left(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w} - y_i\right) = 0$, où $\ell : \mathbb{R} \to \mathbb{R}$ est la fonction de perte de Huber définie par :

$$\ell(t) = \begin{cases} \frac{1}{2}t^2 & \text{if } |t| < 1 \\ |t| - \frac{1}{2} & \text{otherwise.} \end{cases} \tag{3}$$

Cette fonction se comporte comme $t \mapsto \frac{t^2}{2}$ pour $|t| < 1$ et comme $t \mapsto |t|$ lorsque $|t|$ est très grand. Contrairement à ce que son expression peut suggérer, ℓ est continûment dérivable (ou de classe $\mathcal{C}^1$)

L'expression $\ell\left(h_{\boldsymbol{w}}(\boldsymbol{x}_i) - y_i\right)$ représente l'erreur du modèle en $(\boldsymbol{x}_i, y_i)$, et on cherche un modèle (c'est-à-dire un vecteur $\boldsymbol{w} \in \mathbb{R}^d$) tel que la somme de ces erreurs soit minimale. On considère donc :

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} f(\boldsymbol{w}) := \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w} - y_i). \tag{4}$$

a) Justifier que 0 est un minorant de (4). Est-ce sa valeur minimale ?

b) Le gradient de f en $\boldsymbol{w} \in \mathbb{R}^d$ est donné par

$$\nabla f(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} \ell'(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} - y_i) \boldsymbol{x}_i, \tag{5}$$

avec

$$\ell'(t) = \begin{cases} 1 & \text{si} \quad t > 1 \\ t & \text{si} \quad |t| \leq 1 \\ -1 & \text{si} \quad t < -1. \end{cases}$$

Écrire (en pseudo-code) l'itération de descente de gradient avec une taille de pas constante α et en utilisant la formule (5). Que devient cette itération si le point courant est un minimum local ?

c) Une constante de Lipschitz pour ∇f est $L = \frac{1}{n} \sum_{i=1}^{n} \|\boldsymbol{x}_i\|^2$. Comment utiliser cette constante pour définir la longueur de pas ? Lorsque L est inconnue, donner deux choix possibles pour la taille de pas.

d) La fonction f s'écrit $f = \frac{1}{n} \sum_{i=1}^{n} f_i$, où $f_i(\boldsymbol{w}) = \ell(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} - y_i)$. Le gradient f_i en $\boldsymbol{w}$ est

$$\nabla f_i(\boldsymbol{w}) = \ell'(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} - y_i) \boldsymbol{x}_i.$$

Écrire (en pseudo-code) l'itération du gradient stochastique pour ce problème sans choix particulier de taille de pas.

e) *On considère ici que notre unité de coût est un accès à un $\boldsymbol{x}_i$.* Quel est le coût d'une itération de descente de gradient, et celui d'une itération de gradient stochastique ?

f) Quand on applique le gradient stochastique avec une longueur de pas fixe, on peut parfois observer que la méthode génère des itérés de norme de plus en plus grande, ce qui conduit à un dépassement de mémoire pour l'algorithme. Fournir une justification à ce phénomène.

g) On considère une variante par fournées (*batch*) du gradient stochastique, dans laquelle on choisit un sous-ensemble de n_b composantes dans la somme finie de (4).

i) Écrire l'itération correspondante (en pseudo-code).

ii) Si n_b correspond au nombre de processeurs disponibles pour les calculs, quel peut être l'intérêt de choisir n_b comme taille de fournée ?

iii) Donner un autre intérêt plus général des méthodes par fournées en comparaison avec l'algorithme du gradient stochastique basique.

iv) Supposons que l'on utilise plusieurs tailles de fournées et que l'on observe une amélioration en termes de convergence quand n_b augmente pour $1 \leq n_b \leq \frac{n}{10}$. Supposons que l'on observe aussi qu'augmenter n_b au-delà de $n/10$ conduise à une dégradation de la performance. Comment expliquer ces observations ?