Université de Paris
UFR de Mathématiques et Informatique
45, rue des Saints-Pères 75270 Paris cedex 06

MASTER 2 MVA

Modèles stochastiques pour l'analyse d'image

TV ICE - Espérance conditionnelle itérée

A envoyer par courriel avant le 28 février 2020, sous la forme d'un Python Notebook commenté. Utilisez des cellules en markdown pour répondre aux questions. Indiquez votre nom et prénom dans le nom du fichier.

1 Introduction

Ce projet porte sur le débruitage d'image grâce au modèle classique TV-L_2. Il est inspiré de l'article [1]. Supposons que nous observons une image u_0 sur une grille finie Ω, version bruitée d'une image u inconnue

$$u_0 = u + n \tag{1}$$

avec n est un bruit i.i.d. gaussien de variance σ^2.
En supposant un modèle a priori $p(u) \propto e^{-\lambda \mathrm{TV}(u)}$ sur les images, on peut écrire la distribution a posteriori

$$\pi(u) := p(u|u_0) \propto p(u_0|u)p(u) \propto e^{-\frac{1}{2\sigma^2}\|u-u_0\|^2} e^{-\lambda \mathrm{TV}(u)}, \tag{2}$$

avec

$$\mathrm{TV}(u) = \frac{1}{2} \sum_{i\in\Omega, j\in\mathcal{N}(i)} |u(i) - u(j)|,$$

avec $\mathcal{N}(i)$ l'ensemble des 4 voisins directs de i dans la grille Ω et avec la convention que $|u(i)-u(j)| = 0$ si $j \notin \Omega$. Calculer le maximum de cette distribution a posteriori (2) peut se faire très simplement en minimisant $-\log(p(u|u_0))$, ce qui est équivalent à minimiser

$$\frac{1}{2\sigma^2}\|u - u_0\|^2 + \lambda \mathrm{TV}(u),$$

qui est convexe. On peut minimiser cette énergie par exemple grâce à l'algorithme de Chambolle et Pock [2].
Alternativement, on peut calculer la moyenne (l'espérance) de cette distribution a posteriori, qui est l'espérance conditionnelle

$$\mathbb{E}_{U\sim p(u)}[U|u_0] = \int_{\mathbb{R}^{|\Omega|}} u\, p(u|u_0)du. \tag{3}$$

Cette intégrale en grande dimension n'est pas calculable simplement. Une possibilité est d'échantillonner selon la distribution a posteriori grâce à un algorithme de type Markov Chain Monte Carlo et de faire la moyenne des échantillons pour approcher cette espérance conditionnelle, mais la vitesse de convergence est alors en $\frac{1}{\sqrt{n}}$ avec n le nombre d'itérations de l'algorithme.
Dans ce devoir, nous allons étudier une méthode alternative pour calculer une version approchée de cette espérance conditionnelle sans avoir à échantillonner $\pi(u) = p(u|u_0)$.

2 Espérance conditionnelle itérée

Pour n'importe quel pixel i de Ω, notons $i^c = \Omega \setminus \{i\}$. La loi de la valeur $u(i)$ sachant u_0 peut s'écrire comme une marginale de la loi π, donc

$$\pi(u(i)) = \int_{\mathbb{R}^{|\Omega|-1}} \pi(u) du_{|i^c}. \tag{4}$$

S'il est très compliqué de calculer cette loi marginale, la loi marginale conditionnelle $\pi(u(i)|u(i^c))$ est par contre beaucoup plus simple.

▶ **Question 1 :** *Montrez que pour tout pixel i cette loi marginale conditionnelle s'écrit*

$$\pi(u(i)|u(i^c)) \propto e^{-\frac{1}{2\sigma^2}|u(i)-u_0(i)|^2} e^{-\lambda \sum_{j \in \mathcal{N}(i)} |u(j)-u(i)|}. \tag{5}$$

L'idée de l'algorithme TV-ICE (*Iterated Conditional Expectation*) est de remplacer la moyenne a posteriori $\int_{\mathbb{R}^{|\Omega|}} u\pi(u)du$ par des itérations successives des moyennes de toutes ces marginales conditionnelles. L'algorithme consiste donc à itérer

$$\forall i \in \Omega, \;\; u^{k+1}(i) = \mathbb{E}_{U\sim\pi}[U(i)|U(i^c) = u^k(i^c)]. \tag{6}$$

Il est possible de montrer [1] que cet algorithme converge linéairement vers une image asymptotique $\hat{u}_{ICE}$ quelle que soit l'initialisation u^0, et que cette limite satisfait

$$\forall i \in \Omega, \;\; \hat{u}_{ICE}(i) = \mathbb{E}_{U\sim\pi}[U(i)|U(i^c) = \hat{u}_{ICE}(i^c)]. \tag{7}$$

▶ **Question 2 :** *Montrez que pour toute image w de même taille que u_0,*

$$\mathbb{E}_{U\sim\pi}[U(i)|U(i^c) = w(i^c)] = u_0(i) + \frac{1}{Z}\int_{\mathbb{R}} s e^{-\frac{1}{2\sigma^2}s^2 - \lambda \sum_{j\in\mathcal{N}(i)} |s+u_0(i)-w(j)|} ds \tag{8}$$

avec Z une constante à préciser.

3 Algorithme

Pour i fixé, on note $t = u(i)$ et on ordonne les 4 valeurs de $\{w(j)\}_{j\in\mathcal{N}(i)}$, que l'on note $a \le b \le c \le d$. On souhaite donc calculer

$$\mathbb{E}_{U\sim\pi}[U(i)|U(i^c) = w(i^c)] = t + \frac{1}{Z}\int_{\mathbb{R}} s e^{-\frac{1}{2\sigma^2}s^2 - \lambda(|s+t-a|+|s+t-b|+|s+t-c|+|s+t-d|)} ds$$

Cette intégrale se calcule en la découpant sur les intervalles sur lesquels les valeurs absolues sont de signes constants. On admettra qu'on aboutit ainsi à la formule suivante

$$\mathbb{E}_{U\sim\pi}[U(i)|U(i^c) = w(i^c)] = t + 2\sigma^2\lambda \frac{2X_{-2} + X_{-1} - X_1 - 2X_2}{X_{-2} + X_{-1} + X_0 + X_1 + X_2} \tag{9}$$

avec

$$\begin{cases} X_{-2} &= \operatorname{erfc}\left(\frac{t-a+4\lambda\sigma^2}{\sigma\sqrt{2}}\right) \exp\left(2\lambda(2(t+2\lambda\sigma^2)-a-b)\right) \\ X_{-1} &= \left(\operatorname{erf}\left(\frac{b-t-2\lambda\sigma^2}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{a-t-2\lambda\sigma^2}{\sigma\sqrt{2}}\right)\right) \exp\left(\lambda(2(t-b)+2\lambda\sigma^2)\right) \\ X_0 &= \operatorname{erf}\left(\frac{c-t}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{b-t}{\sigma\sqrt{2}}\right) \\ X_1 &= \left(\operatorname{erf}\left(\frac{d-t+2\lambda\sigma^2}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{c-t+2\lambda\sigma^2}{\sigma\sqrt{2}}\right)\right) \exp\left(\lambda(2(c-t)+2\lambda\sigma^2)\right) \\ X_2 &= \operatorname{erfc}\left(\frac{d-t+4\lambda\sigma^2}{\sigma\sqrt{2}}\right) \exp\left(2\lambda(c+d-2(t-2\lambda\sigma^2))\right). \end{cases} \tag{10}$$

où les fonctions erf et erfc sont respectivement la fonction d'erreur $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$ et la fonction d'erreur complémentaire $\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^{+\infty} e^{-t^2}dt$.

Une implémentation naïve de la formule précédente pose des problèmes numériques. Pour les éviter, on écrira l'algorithme itératif TV-ICE sous la forme :

Data : image bruitée u_0, paramètres σ, λ et nombre d'itérations n
Result : estimée u de $\hat{u}_{ICE}$
initialisation $u \leftarrow u^0$;
for $k \leftarrow 1$ **to** n **do**
  for $i \in \Omega$ **do**
    ordonner les valeurs des 4 voisins de i en $a \leq b \leq c \leq d$;
    calculer $(\log X_k)_{-2 \leq k \leq 2}$;
    $M \leftarrow \max_{-2 \leq k \leq 2} \log X_k$;
    $X'_k \leftarrow \exp(\log X_k - M), \quad -2 \leq k \leq 2$;
    $\tilde{u}(i) \leftarrow u_0(i) + 2\lambda\sigma^2 \frac{2X'_{-2}+X'_{-1}-X'_1-2X'_2}{X'_{-2}+X'_{-1}+X'_0+X'_1+X'_2}$
  end
  $u \leftarrow \tilde{u}$
end

Algorithme 1 : Algorithme TV-ICE [1]

▶ **Question 3 :** *Expliquez pourquoi l'algorithme précédent est une bonne solution pour implémenter l'itération* (9)*. Pourquoi faut-il passer au logarithme ? Pour quelle raison faut-il retirer M à toutes les valeurs* $\log X_k$ *? Illustrez l'intérêt de cette manipulation sur un calcul numérique simple.*

▶ **Question 4 :** *Implémentez l'algorithme précédent et testez le pour calculer $\hat{u}_{ICE}$ sur une image de votre choix. Vous pourrez utiliser les fonctions* logerfc *et* logerf2 *déjà fournies. La fonction* logerfc *permet de calculer le logarithme de la fonction d'erreur complémentaire $erfc$. La fonction* logerf2 *permet de calculer le logarithme de $erf(b) - erf(a)$ avec $a < b$. Essayez d'éviter de faire une boucle sur les pixels en vectorisant la mise à jour de l'image. Le nombre total n d'itérations sur toute l'image peut être choisi assez petit en pratique (entre 50 et 100). Vous pouvez choisir des valeurs de σ et λ de la même manière que dans le TP sur l'échantillonnage par exemple.*

▶ **Question 5 :** *Comparez le résultat obtenu avec le Maximum a posteriori (MAP) pour le même problème (vous pouvez le calculer avec l'algorithme de Chambolle-Pock [2] comme vu au premier TP), pour des paramètres équivalents. Commentez.*

▶ **Question 6 :** *Montrez expérimentalement que la vitesse de convergence des itérées u^k vers leur limite u^∞ est linéaire, c'est-à-dire que*

$$\|u^k - u^\infty\|_2 = O(\alpha^n)$$

avec $0 \leq \alpha < 1$.

Références

[1] Louchet, C., Moisan, L. *Total variation denoising using iterated conditional expectation.* In 2014 22nd European Signal Processing Conference (EUSIPCO) (pp. 1592-1596). 2014.

[2] Chambolle, A., Pock, T.. *A first-order primal-dual algorithm for convex problems with applications to imaging.* Journal of mathematical imaging and vision, 40(1), 120-145, 2011.