Vers une méthode d'optimisation non paramétrique pour l'apprentissage de dictionnaire en utilisant Small-Variance Asymptotics pour modèle probabiliste

Hong-Phuong DANG[1], Clément ELVIRA[2], et Pierre CHAINAIS[3]

[1]ENSAI Bruz, CREST - CNRS UMR 9194
[2]Université Rennes, INRIA, IRISA - UMR 6074
[3]Université Lille, Centrale Lille, CRIStAL - UMR 9189

12 juin 2018

Résumé

Dans les modèles probabilistes, les méthodes d'échantillonnage et les approximations variationnelles sont souvent utilisées pour l'inférence. Mais ces méthodes passent difficilement à l'échelle. Une alternative consiste à relâcher le modèle probabiliste dans une formulation non probabiliste et utiliser un algorithme évolutif pour résoudre le problème de minimisation associé. Nous proposons une analyse dite de *Small-Variance Asymptotics* (SVA) du modèle bayésien non paramétrique de type Buffet Indien à deux paramètres qui apprend automatiquement un dictionnaire de taille adaptée. Cette approche s'obtient en faisant tendre la variance de la vraisemblance du modèle vers 0. L'analyse montre une interaction entre les méthodes bayésiennes et optimisation. Les résultats illustrent la pertinence de la méthode proposée.

Mots-clef : Bayésien non paramétrique (BNP), Small-Variance Asymptotics (SVA), processus du Buffet Indien (IBP), représentation parcimonieuse, apprentissage de dictionnaire (AD), problème inverse.

1 Introduction

De nombreux problèmes d'apprentissage sont résolus avec des modèles à variables latentes. Le choix du nombre de ces variables latentes conditionne la qualité des estimateurs. La sous(sur)-estimer peut causer du *sous(sur)-apprentissage*. Une approche classique consiste à pénaliser la vraisemblance du modèle par le nombre de variables latentes, avec par exemples des critères de type AIC, BIC, *etc.*

À l'inverse, la modélisation *bayésienne non paramétrique* (BNP) intègre le nombre de variables latentes au modèle. Cette quantité devient une variable aléatoire, et est estimée conjointement avec les variables latentes. Ainsi les modèles BNP peuvent être interprétés comme des modèles paramétriques avec un nombre infini de paramètres, mais où seul un nombre presque surement fini sont actifs. Les méthodes traditionnellement utilisées pour l'inférence reposent sur des algorithmes MCMC et/ou des approximations variationnelles. Mais les méthodes exactes sont coûteuses et passent difficilement à l'échelle, et les approximations se ramènent pour l'instant à des modèles paramétriques.

L'approche *Small-Variance Asymptotics* (SVA) offre des techniques utiles pour établir des liens conceptuels entre des modèles probabilistes et non probabilistes et dériver de nouveaux algorithmes simples et évolutifs [1, 2]. Elle a été appliquée avec succès aux problèmes de traitement du signal [3]. Cette approche permet d'aller vers une méthode qui profite à la fois du côté non paramétrique des méthodes probabilistes et l'avantage numérique des méthodes d'optimisation. Nous proposons dans ce travail une approche SVA pour l'*apprentissage de dictionnaire* (AD) [4]. Le modèle s'appuie sur un *processus du buffet Indien* (IBP) pour associer chaque donnée à un petit nombre d'éléments du dictionnaires, qui constituent les variables latentes.

La partie 2 rappelle le principe de l'AD. La partie 3 présente l'extension du processus Buffet Indien et le modèle proposé. La partie 4 décrit l'analyse de SVA sur le modèle proposé ainsi que l'algorithme pour l'inférence. La partie 5 est consacrée aux résultats expérimentaux et à la discussion.

2 Apprentissage de dictionnaire

En traitement d'image, l'AD pour la représentation parcimonieuse est bien connue dans le cadre de la résolution de problèmes inverses mal posés [4]. Le problème est souvent présenté sous la forme :

$$\mathbf{Y} = \mathbf{H}(\mathbf{X} + \boldsymbol{\varepsilon}) \quad \text{où} \quad \mathbf{X} = \mathbf{DW} \tag{1}$$

$\mathbf{Y} \in \mathbb{R}^{L \times N}$ est un ensemble de N observations $\mathbf{y}_i$. Chaque vecteur colonne $\mathbf{y}_i \in \mathbb{R}^L$ représente une imagette (*patch*, par ex. 8×8, donc L=64), par ordre lexicographique. $\mathbf{X} \in \mathbb{R}^{L \times N}$ représente les patches extraits à partir de l'image initiale qui sont perturbés par un bruit $\boldsymbol{\varepsilon}$ et un opérateur linéaire $\mathbf{H}$ connu. $\mathbf{H}$ peut être la matrice identité, il s'agit d'un problème de débruitage. Si $\mathbf{H}$ est l'ensemble des matrices diagonales binaires, on est dans le cas de l'inpainting. L'AD peut être abordé sous l'angle de la factorisation de matrice où l'on décompose $\mathbf{X}$ sous la forme du produit de deux matrices. $\mathbf{D}=[\mathbf{d}_1, ..., \mathbf{d}_K] \in \mathbb{R}^{L \times K}$ est un dictionnaire redondant où le nombre d'atomes K est supérieur à la dimension de l'espace dans lequel vivent les données. $\mathbf{W}=[\mathbf{w}_1, ..., \mathbf{w}_i] \in \mathbb{R}^{K \times N}$ est la matrice des cœfficients. Chaque vecteur colonne $\mathbf{w}_i$ encode la représentation parcimonieuse de chaque observation $\mathbf{x}_i$ sur $\mathbf{D}$. Inférer $\mathbf{X}$ est équivalent ici à la recherche d'un couple optimal ($\mathbf{D}$,$\mathbf{W}$) à partir de $\mathbf{Y}$.

La parcimonie est typiquement imposée par l'ajout d'une pénalité ℓ_p, $p \in \{0, 1\}$ dans un problème d'optimisation jointe (d'autres formulations sont possibles)

$$(\mathbf{D}, \mathbf{W}) = \underset{(\mathbf{D},\mathbf{W})}{\operatorname{argmin}} \quad \frac{1}{2}\|\mathbf{H}(\mathbf{Y} - \mathbf{DW})\|_2^2 + \lambda \|\mathbf{W}\|_p \tag{2}$$

Dans la littérature, la taille de dictionnaire est le plus souvent fixée à l'avance à $K = 256$ ou 512 atomes [4, 5].

Dans le cadre bayésien, la vraisemblance est construite conformément au modèle (1) où le bruit est souvent gaussien i.i.d.. La loi *a priori* $\mathrm{p}(\mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}})$ joue le rôle de régularisation. Le problème s'écrit typiquement sous la forme d'une loi jointe *a posteriori* :

$$\mathrm{p}(\mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}} | \mathbf{Y}, \mathbf{H}) \propto \mathrm{p}(\mathbf{Y} | \mathbf{H}, \mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}}) \mathrm{p}(\mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}}) \tag{3}$$

En utilisant par exemple l'échantillonnage de Gibbs pour l'inférence, le problème peut être résolu en échantillonnant alternativement $\mathbf{D}$, $\mathbf{W}$ et $\sigma_{\boldsymbol{\varepsilon}}$. Dans un cadre BNP, le dictionnaire est appris sans fixer sa taille au préalable et aucun réglage de paramètre n'est nécessaire. Le modèle IBP-DL [6] utilise le processus du buffet indien comme loi *a priori* pour favoriser la parcimonie et contrôler la taille du dictionnaire.

3 Extension du modèle IBP-DL

3.1 Processus du Buffet Indien (IBP)

Conformément à l'équation (1), le nombre d'atomes est non seulement défini par le nombre de colonnes de $\mathbf{D}$ mais aussi par le nombre de lignes de $\mathbf{W}$. Lorsqu'une ligne de $\mathbf{W}$ ne contient que des zéros, cela revient à supprimer de $\mathbf{D}$ la colonne de l'atome correspondant à cette ligne de $\mathbf{W}$. Le support de la matrice binaire d'affectations $\mathbf{Z}$ modélise l'utilisation parcimonieuse des éléments d'un dictionnaire par des données dans un modèle à caractéristiques latentes (*Latent Features*). Si la donnée i est associée à la caractéristique k alors $\mathbf{Z}(k, i)$=1, (0 si non). Lorsque le nombre de caractéristiques utilisées est inconnu, l'IBP [7] permet de définir une loi *a priori* sur $\mathbf{Z}$ de taille (nombre de lignes) potentiellement infinie. Il permet donc de ne pas fixer à l'avance le nombre d'atomes (caractéristiques) K et pénalise simultanément la valeur de K. L'IBP émule une distribution échangeable sur des matrices binaires *creuses* et *potentiellement infinies*.

Une extension de l'IBP à deux paramètres [8] est présentée. Son processus génératif est le suivant. N clients (données) prennent des plats (caractéristiques) dans un buffet (Indien) potentiellement infini. Chaque client $i+1$ commence par se servir dans les plats déjà entamés avec probabilité $\frac{m_k}{i+c}$, où m_k est le nombre de clients précédents ayant choisi le plat k. Puis, il essaie Poisson($\frac{\alpha c}{i+c}$) nouveaux plats. Cette étape permet d'enrichir progressivement le dictionnaire. La distribution de probabilité correspondante sur les classes d'équivalence $[\mathbf{Z}]$, quand $K \to \infty$, est donnée par

$$\mathrm{P}([\mathbf{Z}]) = \frac{(\alpha c)^K}{\prod_{h=1}^{2^N - 1} K_h!} \exp(-\alpha c H_N) \prod_{k=1}^{K} \beta(m_k, N - m_k + c) \tag{4}$$

où $H_N = \sum_{j=1}^{N} (c + j - 1)^{-1}$, β indique la fonction Bêta, N le nombre de données ou encore le nombre de colonnes de $\mathbf{Z}$, $m_k = \sum_{i=1}^{N} \mathbf{Z}(k, i)$ le nombre de données utilisant l'atome k, K le nombre d'atomes "actifs" tels que m_k>0, K_h est le nombre d'atomes avec le même *histoire* $\mathbf{Z}(k, :)$=h. Autrement dit, les plats ont été choisis par le même ensemble de clients. Le paramètre α contrôle le nombre total K de caractéristiques et le paramètre de concentration c régularise la parcimonie. En fixant c=1, on retrouve l'IBP usuel à un paramètre [7]. L'espérance du nombre de caractéristiques K est $\mathbb{E}[K] = \alpha c H_N \approx \alpha c \log(N)$.

3.2 Généralisation du modèle IBP-DL

Dans [6], un modèle BNP pour l'AD (IBP-DL) a été proposé en utilisant l'IBP usuel comme la loi *a priori*

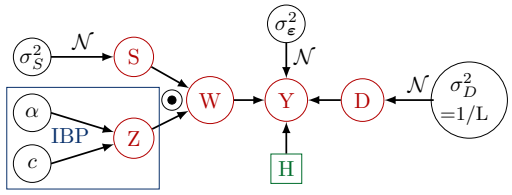

FIGURE 1 – Modèle graphique de IBP-DL.

sur le support de matrice des cœfficients afin de promouvoir la parcimonie. L'acronyme IBP-DL signifie Indian Buffet Process for Dictionary Learning. Dans cet article, nous explorons l'intérêt de la généralisation du modèle IBP-DL à deux paramètres. La figure 1 montre le modèle graphique de l'IBP-DL. Le modèle peut être décrit par : $\forall i \in [\![1, N]\!]$

$$\mathbf{y}_i = \mathbf{H}_i[(\mathbf{D}\mathbf{w}_i) + \boldsymbol{\varepsilon}_i], \text{ with } \mathbf{w}_i = \mathbf{z}_i \odot \mathbf{s}_i, \quad (5)$$

$$\mathbf{Z} \sim \text{IBP}(\alpha, c), \quad s_{ki} \sim \mathcal{N}(0, \sigma_s^2), \forall k \in \mathbb{N}, \quad (6)$$

$$\mathbf{d}_k \sim \mathcal{N}(0, \sigma_D^2 \mathbb{I}_L), \forall k \in \mathbb{N}, \quad (7)$$

$$\mathbf{H}_i\boldsymbol{\varepsilon}_i \sim \mathcal{N}(0, \sigma_{\boldsymbol{\varepsilon}}^2 \mathbb{I}_L). \quad (8)$$

$\odot$ est le produit Hadamard (terme à terme). Dans l'esprit d'un modèle Bernoulli-gaussien paramétrique, la parcimonie de $\mathbf{W}$ est induite par celle de matrice binaire $\mathbf{Z}$ grâce à l'*a priori* non paramétriques IBP. Cette loi contrôle le nombre de lignes de $\mathbf{Z}$ (ou encore $\mathbf{W}$) qui représente aussi le nombre d'atomes de $\mathbf{D}$. L'amplitude du code $\mathbf{W}$ est donnée par $\mathbf{S}$. Si $\mathbf{Z}(k,i) = 1$, la donnée i utilise l'atome k alors $\mathbf{W}(k,i) = \mathbf{S}(k,i)$, (0 si non). Des lois *a priori* conjuguées Gamma et inverse Gamma sont utilisées pour les autres paramètres [6]. Seule la variance $\sigma_D^2{=}1/L$ pour régler le problème d'indétermination liée à la norme du couple ($\mathbf{D}$, $\mathbf{W}$). Notons que cela revient à écrire que l'énergie contenue dans chaque atome k vaut, en espérance, $\mathbb{E}[\mathbf{d}_k^{\mathrm{T}}\mathbf{d}_k] = \sum_{i=1}^{L} 1/L = 1$. Il s'agit donc d'une manière douce de les normaliser.

4 Méthode proposée

Cette partie présente IBPDL-SVA, une approche numériquement efficace pour approcher des estimateurs bayésiens basée sur l'approximation SVA. Les liens conceptuels entre des approches non probabilistes et IBPDL-SVA seront discutés après.

4.1 Small Variance Asymptotic (SVA)

Dans [6], un algorithme utilisant l'échantillonneur de Gibbs avec Métropolis-Hasting a été proposé pour échantillonner la distribution jointe *a posteriori* :

$$\mathrm{p}(\mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}}^2, \ldots | \mathbf{Y}, \mathbf{H}) \propto \mathrm{p}(\mathbf{Y}|\mathbf{H}, \mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}}^2)\mathrm{p}(\mathbf{D})\mathrm{p}(\mathbf{W})\mathrm{p}(\sigma_{\boldsymbol{\varepsilon}}^2). \quad (9)$$

Mais cet algorithme est coûteux en temps calcul et passe difficilement à l'échelle. Nous étudions ici une alternative basée sur une analyse SVA. L'analyse SVA nécessite d'abord de construire l'estimateur approché MAP asymptotique (aMAP) de (9) selon

$$\widehat{\mathbf{D}}, \widehat{\mathbf{W}} = \underset{\mathbf{D},\mathbf{W}}{\operatorname{argmin}} \lim_{\sigma_{\boldsymbol{\varepsilon}}^2 \to 0} -2\sigma_{\boldsymbol{\varepsilon}}^2 \log \mathrm{p}(\mathbf{D}, \mathbf{W}, \sigma_{\boldsymbol{\varepsilon}}^2 | \mathbf{Y}, \mathbf{H}). \quad (10)$$

Tel quel, l'aMAP est l'estimateur du maximum de vraisemblance. Comme indiqué dans [9], il est nécessaire de coupler les hyperparamètres du modèle pour les faire évoluer avec $\sigma_{\boldsymbol{\varepsilon}}^2$ et conserver la propriété de régularisation désirée du modèle bayésien. Posons $\alpha = \exp\left(\frac{\sigma_{\boldsymbol{\varepsilon}}^2}{\lambda_1} - \frac{\lambda_1}{2\sigma_{\boldsymbol{\varepsilon}}^2}\right)$, $c = \exp\left(\frac{\lambda_2}{2\sigma_{\boldsymbol{\varepsilon}}^2} - \frac{\sigma_{\boldsymbol{\varepsilon}}^2}{\lambda_2}\right)$ et $\lambda_1,\ \lambda_2 > 0$. En laissant $\sigma_{\boldsymbol{\varepsilon}}^2 \to 0$, on trouve asymptotiquement :

$$-2\sigma_{\boldsymbol{\varepsilon}}^2 \log \mathrm{p}(\mathbf{Y}, \mathbf{H}, \mathbf{D}, \mathbf{W}) \sim \sum_{N}^{i=1} [\mathbf{y}_i - \mathbf{H}_i(\mathbf{D}\mathbf{w}_i)]^{\mathrm{T}}[\mathbf{y}_i - \mathbf{H}_i(\mathbf{D}\mathbf{w}_i)] + \lambda_2 \sum_{k}^{K} m_k + (\lambda_1 - \lambda_2)(K+1) \quad (11)$$

où m_k est le nombre de données utilisant l'atome k. La somme des produits scalaires provient de la fonction exponentielle de la vraisemblance gaussienne, et le terme de pénalité provient de la loi *a priori* IBP. Il est à noter que cette somme revient à la somme des normes ℓ_2 $\|.\|_2^2$ et $\sum_{k=1}^{K} m_k = \|\mathbf{W}\|_0$.

L'équation (11) montre que l'estimateur aMAP du problème d'AD est asymptotiquement équivalent à la solution du problème d'optimisation suivant :

$$\underset{K,D,W}{\operatorname{argmin}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{H}_i(\mathbf{D}\mathbf{w}_i)\|_2^2 + \lambda_2 \|\mathbf{W}\|_0 + (\lambda_1 - \lambda_2)(K+1). \quad (12)$$

En comparant avec la fonction coût (2) du problème d'optimisation standard, l'équation (12) contient non seulement un terme de pénalité sur la parcimonie mais aussi un terme de pénalité sur le nombre d'atomes. La parcimonie est contrôlée par λ_2 tandis que λ_1 contrôle le nombre d'atomes. Dans [2], une fonction similaire à (12) sans le terme ℓ_0 est désignée comme la fonction objectif de *BP-means* (BP pour *Beta process*). L'apparition de cette pénalité ℓ_0 provient du second paramètre de l'IBP et promeut la parcimonie, ce qui n'était pas le cas dans [2]. En plus, le changement de variable $\alpha = f(\lambda_1)$ est différent pour correspondre à tout le domaine de α.

4.2 Algorithme IBPDL-sva

L'approche SVA suggère de concevoir un algorithme déterministe basé sur le comportement asymptotique d'un échantillonneur de Gibbs [9, 2]. L'algorithme 1 résume une optimisation alternée sur $\mathbf{D}$ et $\mathbf{W}$.

Entrées : $\mathbf{Y}$, T, λ_1, λ_2
Sorties : dictionnaire $\mathbf{D}$, des cœfficients $\mathbf{W}$
$\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_i] \leftarrow \mathbf{Y}$
Pour *itération t=1 :T*
  Pour *donnée i=1 :N*
    Encodage : OMP
    $k \leftarrow 0$
    Tant que *Vraie*
      $k \leftarrow k+1$
      $\boldsymbol{r}_i \leftarrow \mathbf{y}_i, \mathbf{w}_i \leftarrow [0, \ldots, 0]$
      $k^* = \text{argmax}|\langle \boldsymbol{r}_i, \mathbf{d}_k \rangle|$ $\quad \mathcal{O}(K-k+1)$
      Trouver $\mathbf{w}^*$ avec MC[1] $\quad \mathcal{O}(k^2L + k^3)$
      Si $\mathbf{w}^*$ *augmente eq.* (12)
        break
      fin
    fin
    Ajout d'atomes
    Proposer $\mathbf{d}_n$ selon (16)
    Si $\mathbf{d}_\text{n}$ *diminue eq.* (12)
      $\mathbf{D} = [\mathbf{D}, \mathbf{d}_\text{n}]$
      Recalculer $\mathbf{w}_i$ $\quad \mathcal{O}(K^2L + K^3)$
    fin
    Mise à jour du dictionnaire
    Retirer les atomes inutilisés
    Mettre à jour $\mathbf{D}$ selon (17) ou (18)
  fin
fin

Algorithm 1: Pseudo-algorithme de IBPDL-SVA ([1] Moindres Carrés).

Encodage sur les atomes actifs : Soit le vecteur résidu $\boldsymbol{r}_{k,i} = \mathbf{y}_i - \mathbf{H}_i \sum_{j \neq k} w_{j,i}\mathbf{d}_j$. La distribution *a posteriori* $\text{p}(\mathbf{w}_{k,i}|\mathbf{Y}) \propto \sum \text{p}(s_{k,i}|\mathbf{Y}, z_{k,i} = z)\text{P}[z_{k,i} = z|\mathbf{Y}] \propto p_0 + p_1$ avec $z \in \{0, 1\}$ peut être marginalisée par rapport à $s_{k,i}$. Quand $\sigma_{\boldsymbol{\varepsilon}}^2 \to 0$, on obtient

$$\log p_0 = \|\boldsymbol{r}_{k,i}\|_2^2 + \lambda_2 m_k \tag{13}$$

$$\log p_1 = \|\boldsymbol{r}_{k,i} - \mathbf{d}_k^\text{T}\boldsymbol{r}_{k,i}\|_2^2 + \lambda_2(m_k + 1). \tag{14}$$

La variable aléatoire de Bernoulli $z_{k,i}$ de paramètre p_{ki} indique si l'observation $\mathbf{y}_i$ est décrite par $\mathbf{d}_k$. Cela suggère de mettre $z_{k,i} = 1$ si $p_{ki} \geq .5$ et 0 sinon. On voit immédiatement à partir de (13) et (14) que $p_{ki} \geq .5$ ssi $w_{k,i} = \mathbf{d}_k^\text{T}\boldsymbol{r}_{k,i}$ diminue la fonction de coût (12) par rapport à $w_{k,i} = 0$.

Dans un échantillonneur de Gibbs, on fait une boucle sur un ordre aléatoire de $k \in \{1, \ldots, K\}$. Pour éviter ce recourt à l'aléatoire, nous proposons plutôt de mettre $\mathbf{w}_{1:K,i} = 0$ et de commencer par les atomes les plus corrélés avec l'erreur résiduelle. Cette procédure ressemble à l'algorithme de Matching Pursuit (MP) [10] avec la fonction de coût comme critère d'arrêt. Nous avons choisi de remplacer MP par Orthonormal Matching Pursuit (OMP) [11] qui offre de meilleures performances d'estimation à un coût raisonnable.

Ajout d'un nouvel atome : on teste si l'ajout d'un nouvel atome permet une meilleure reconstruction. Afin de respecter les hypothèses d'un échantillonnage de Gibbs correct, Metropolis-Hasting a été utilisé dans [6]. Le choix de la loi de proposition est libre puisque la proposition est corrigée par un rapport de distribution de probabilité. Ici, nous choisissons simplement d'explorer l'espace autour du vecteur résidu, c.-à-d. que la loi de proposition $\text{q}(\mathbf{d}_\text{n}|\mathbf{Y}, \mathbf{W}, \mathbf{D})$ d'un nouvel atome $\mathbf{d}_\text{n}$ est Normal de paramètres

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathbf{d}_\text{n}} &= \left(\sigma_D^{-2}\mathbb{I}_L + \sigma_{\boldsymbol{\varepsilon}}^{-2}\sum_{i=1}^{N} s_{\text{new},i}^2 \mathbf{H}_i\right)^{-1} \\ \boldsymbol{\mu}_{\mathbf{d}_\text{n}} &= \sigma_{\boldsymbol{\varepsilon}}^{-2}\boldsymbol{\Sigma}_{\mathbf{d}_\text{n}} \sum_{i=1}^{N} s_{\text{new},i}(\mathbf{y}_i - \mathbf{H}_i \sum_{k=1}^{K} \mathbf{d}_k w_{ki}) \end{aligned} \tag{15}$$

Quand les $s_{\text{new},i}$ sont marginalisés et $\sigma_{\boldsymbol{\varepsilon}}^2 \to 0$, la gaussienne se réduit à un Dirac, suggérant la proposition :

$$\mathbf{d}_\text{n} = \sum_{i=1}^{N} \frac{1}{N}(\mathbf{y}_i - \mathbf{H}_i \sum_{k}^{K} \mathbf{d}_k \mathbf{w}_{ki}). \tag{16}$$

L'équation (16) correspond à une approximation de rang 1 des résidus. Notons que l'approximation optimale de rang 1 au sens des moindres carrés est le vecteur propre associé à la valeur propre la plus élevée, correspondant à une mise à jour de type K-SVD [5]. À cause de la complexité numérique, ce choix n'est pas adopté ici. Même si l'approximation a été choisie non optimale, l'exécution de l'argument limitant SVA dans un simple échantillonneur de Gibbs a naturellement conduit à un algorithme qui peut être interprété comme une version non paramétrique de K-SVD [5]. Dans [6], le nouvel atome $\mathbf{d}_\text{n}$ est marginalisé au lieu des poids $s_{\text{new},i}$, permettant un estimateur à plus faible variance. Une telle stratégie n'a pas d'équivalent dans un cadre d'optimisation mais pourrait être importée grâce à l'analyse SVA. Ce travail est en cours d'investigation.

Mise à jour D : on peut utiliser l'espérance de la loi *a posteriori* de $\mathbf{D}$ en laissant $\sigma_{\boldsymbol{\varepsilon}}^2 \to 0$. Nous avons choisi de conserver la partie de correction de bruit pour des raisons de stabilité numérique. Ce choix conduit à deux possibilités :
Mettre à jour colonne par colonne (atome)

$$\mathbf{D}(:,k) = \left(\frac{\bar{\sigma}_{\boldsymbol{\varepsilon}}^2}{\sigma_D^2}\mathbb{I}_L + \sum_{i=1}^{N} w_{ki}^2 \mathbf{H}_i\right)^{-1} \sum_{i=1}^{N} w_{ki}(\mathbf{y}_i - \mathbf{H}_i \sum_{j \neq k}^{K} \mathbf{d}_j w_{ji}) \tag{17}$$

Mettre à jour ligne par ligne (emplacement des pixels)

$$\mathbf{D}(\ell,:) = \mathbf{Y}(\ell,:)\mathbf{W}^\text{T}\left(\mathbf{W}\mathbf{F}_\ell\mathbf{F}_\ell^\text{T}\mathbf{W}^\text{T} + \frac{\bar{\sigma}_{\boldsymbol{\varepsilon}}^2}{\sigma_D^2}\mathbb{I}_K\right)^{-1} \tag{18}$$

	$\sigma_\varepsilon = 25$ PSNR $\approx$ 20.14 dB			$\sigma_\varepsilon = 40$ PSNR $\approx$ 16.06 dB		
Barbara	28.28 K=80	29.06 28.82	27.84 30.72	25.76 K=71	26.34 25.60	25.17 27.99
Hill	28.65 K=63	28.80 28.58	28.51 29.85	27.29 K=14	26.93 26.29	26.80 27.99
Mandrill	24.29 K=148	24.59 24.88	23.58 27.85	22.25 K=61	22.29 22.43	21.71 25.37
Lena	30.49 K=74	31.12 30.45	28.86 32.08	28.81 K=24	28.78 27.58	26.74 29.86
Peppers	30.25 K= 88	29.64 30.23	28.87 30.16	28.23 K=13	27.06 27.27	26.66 27.70

TABLE 1 – Résultats de débruitage sur 5 images pour 2 niveaux de bruit. Pour chaque image, à gauche le PSNR et la taille du dictionnaire issus d'IBPDL-SVA, au centre les PSNRs issus d'IBPDL-Gibb (en haut) et DLENE (en bas), à droite K-SVD (256 atomes, en haut) et BM3D (en bas).

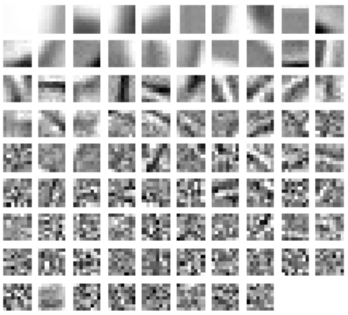

FIGURE 2 – Dictionnaire de Peppers (σ_ε=25) issu de IBPDL-SVA avec K=88 atomes.

σ_ε Missing	$\sigma_\varepsilon = 15$		$\sigma_\varepsilon = 25$	
80%	24.95 K=12	25.28 25.17	23.49 K=24	23.74 23.49
50%	28.58 K=52	28.90 29.31	26.69 K=47	26.54 26.79
20%	30.19 K= 43	30.68 29.93	28.30 K= 65	28.10 27.58

TABLE 2 – Résultats de l'inpainting en présence de bruit sur un segment 256 × 256 de *Barbara* en niveau de gris. Pour chaque case : à gauche le PSNR et la taille du dictionnaire issus d'IBPDL-SVA, à droite les PSNRs issus d'IBPDL-Gibbs (en haut) et de BPFA (en bas).

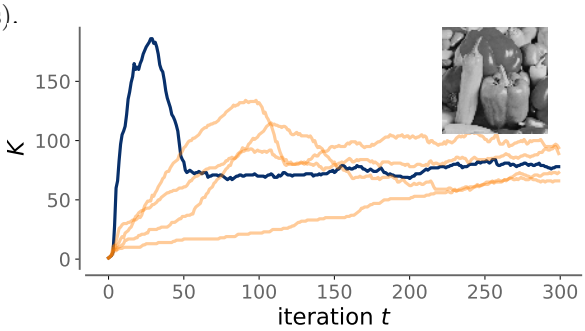

FIGURE 3 – Evolution de la taille du dictionnaire à travers les itérations pour plusieurs couples de (λ_1, λ_2). Les coubes orange représentent les couples éliminés, la courbe bleue est celle retenue.

où $\bar{\sigma}_\varepsilon^2$ est la variance du résidu à l'itération $t-1$. $\mathbf{H}_i$ est une matrice binaire diagonale de taille $L \times L$ où les zéros indiquent les pixels manquants sur le patch i. $\mathbf{F}_\ell$ est une matrice binaire diagonale de taille N. $\mathbf{F}_\ell(i,i)$ indique si le pixel à l'emplacement ℓ du patch i est observé ou non, ainsi $\mathbf{F}_\ell(i,i)$=$\mathbf{H}_i(\ell,\ell)$.

5 Résultats numériques

Cette partie décrit une brève expérience pour montrer que IBPDL-SVA peut bénéficier de certaines propriétés des techniques bayésiennes tout en offrant la rapidité et l'évolutivité des méthodes déterministes.

La première expérience illustre la pertinence du dictionnaire inféré avec IBPDL-SVA en comparant ses performances de débruitage avec 1) IBP-DL avec l'échantillonneur de Gibbs [6] 2) DLENE [12] 3) K-SVD avec K=256 [5] 4) BM3D [13] comme référence de l'état de l'art. Les résultats de BM3D sont rappelés pour information seulement puisque nous ne nous attendons pas à obtenir de meilleures performances. 5 images de taille 512×512 sont traitées, pour 2 niveaux de bruit σ_ε=25 ou 40. Chaque image est formée de 255025 patches (8×8) qui se chevauchent, mais l'approche proposée est testée avec N=16129 soit 50% de chevauchement. On utilise la même base de données réduite pour K-SVD et DLENE. K-SVD trouve un dictionnaire optimal selon le critère de parcimonie retenu pour une image. DLENE adapte la taille du dictionnaire en visant un compromis entre l'erreur de reconstruction et la parcimonie. Le choix du couple (λ_1, λ_2) sera discuté plus bas.

La table 1 affiche les résultats. Comme prévu, les performances de IBPDL-SVA sont inférieures à celles de BM3D mais similaires à celles d'IBPDL-Gibbs et de DLENE, et meilleures que K-SVD. Remarquons la sensibilité de K-SVD à l'ensemble d'apprentissage. Les mêmes conclusions peuvent être tirées par rapport à

IBPDL-Gibbs : la taille K du dictionnaire de l'IBPDL-SVA est inférieure à 256, la valeur utilisée pour K-SVD. De plus, dans le cas d'un bruit fort (σ=40), K est souvent plus petit que 64 qui n'est pas toujours un dictionnaire redondant mais la performance de débruitage reste comparable.

Dans ce papier, le couple (λ_1, λ_2) est obtenue par validation croisée sur Peppers. Nous avons trouvé $(0.12, 0.08)$ pour σ_ε=25 et $(0.4, 0.2)$ pour σ_ε=40. La Fig. 2 illustre le dictionnaire appris par IBPDL-SVA sur l'image Peppers bruitée à $\sigma_{\boldsymbol{\varepsilon}}$=25. Les simulations sont exécutées sur un ordinateur portable personnel et une implémentation Python. La Fig. 3 montre la convergence de la taille K du dictionnaire à travers les itérations pour Peppers avec $\sigma_{\boldsymbol{\varepsilon}}$=25 pour plusieurs couples de (λ_1, λ_2). Cette évolution s'applique à toutes les chaînes : la méthode commence par ajouter trop d'atomes avant de se stabiliser après une période de chauffe. L'AD lié à la courbe bleue coûte environ 30 minutes pour 150 itérations. À titre de comparaison, IBP-DL avec l'échantillonnage de Gibbs a besoin de 1 heure pour 30 itérations avec une implémentation Matlab. Notons qu'optimiser la valeur (λ_1, λ_2) permet de meilleures performances. Cette observation motive un futur travail concernant l'estimation conjointe de ces hyperparamètres via des approches bayésiennes.

La Table 2 montre la pertinence de IBPDL-SVA à travers les résultats d'inpainting. Ici, (λ_1, λ_2) est obtenu par validation croisée sur l'image avec 80% de donnée manquante. IBPDL-SVA est comparé avec IBPDL-Gibbs et BPFA [14]. BPFA est une approche de la famille bayésienne basée sur un a priori Beta-Bernoulli mais fonctionne avec un nombre d'atomes fixé malgré des connexions avec les approches BNP. À nouveau, les performances de IBPDL-SVA sont comparables à celles des deux autres approches.

6 Conclusion

Cet article présente une nouvelle approche numériquement efficace pour l'apprentissage de dictionnaire en utilisant une analyse Small Variance Asymptotic sur un modèle bayésien non paramétrique. L'approche proposée conserve certains des avantages du BNP tels que l'inférence d'un dictionnaire de taille inconnue. Ceci décrit les connexions qui apparaissent entre le comportement asymptotique des approches MCMC et les algorithmes bien connus pour l'AD. Les performances de débruitage de IBPDL-SVA illustrent la pertinence des dictionnaires appris. De futurs travaux visent à utiliser les avantages du cadre bayésien pour inférer les hyperparamètres λ_1, λ_2 et ainsi donner naissance à une approche rapide complètement non supervisée. D'autres applications de BNP telles que PCA non paramétrique [15] peuvent être revisitées.

Références

[1] K. Jiang, B. Kulis, and M. I. Jordan, "Small-variance asymptotics for exponential family dirichlet process mixture models," in *NIPS*, 2012.

[2] T. Broderick, B. Kulis, and M. Jordan, "Mad-bayes : Map-based asymptotic derivations from bayes," in *ICML*, 2013.

[3] M. Pereyra and S. McLaughlin, "Fast unsupervised bayesian image segmentation with adaptive spatial regularisation," *IEEE Trans. Ima. Process.*, 2017.

[4] I. Tosic and P. Frossard, "Dictionary learning : What is the right representation for my signal," *IEEE Signal Process. Magazine*, 2011.

[5] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Sig. Process.*, 2006.

[6] H.-P. Dang and P. Chainais, "Indian buffet process dictionary learning : algorithms and applications to image processing," *Int. J. of Approx. Reasoning*, 2017.

[7] T. L. Griffiths and Z. Ghahramani, "Infinite latent feature models and the indian buffet process," in *NIPS*, 2006.

[8] Z. Ghahramani, T. L. Griffiths, and P. Sollich, "Bayesian nonparametric latent feature models," *Bayesian Statistic*, 2007.

[9] B. Kulis and M. I. Jordan, "Revisiting k-means : New Algorithms via Bayesian Nonparametrics," in *ICML*, 2012.

[10] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Sig. Process.*, 1993.

[11] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit : Recursive function approximation with applications to wavelet decomposition," *Asilomar*, 1993.

[12] M. Marsousi, K. Abhari, P. Babyn, and J. Alirezaie, "An adaptive approach to learn overcomplete dictionaries with efficient numbers of elements," *IEEE Trans. Sig. Process.*, 2014.

[13] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Trans. Ima. Process.*, 2007.

[14] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin, "Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images," *IEEE Trans. Ima. Process.*, 2012.

[15] C. Elvira, P. Chainais, and N. Dobigeon, "Bayesian nonparametric principal component analysis," *preprint arXiv*, 2017.