

Segmentation d'images hyper-spectrales

Hyper-spectral images segmentation

Robin Girard, robin.girard@imag.fr

Manuscrit reçu le 19 décembre 2005

Résumé et mots clés

Nous présentons un algorithme de segmentation d'images hyper-spectrales supposées constantes par régions. Cet algorithme est composé de trois phases. La première est une phase de débruitage inspirée d'une méthode adaptative de lissage pondéré fondée sur une segmentation par croissance de régions, la deuxième se sert des paramètres obtenus lors du débruitage de la première phase et a pour but de produire une estimation des contours des régions connexes issues du débruitage. La dernière étape de l'algorithme consiste à fusionner les régions issues de la deuxième phase en minimisant une version pénalisée de l'erreur quadratique de reconstruction. La méthodologie est illustrée sur des données simulées d'imagerie de résonance magnétique nucléaire.



Abstract and key words

We present a new image segmentation algorithm for hyper-spectral images that are supposed to be piecewise constant. The procedure is composed by three steps. The first step (denoising) is inspired by a nonparametric adaptive weights smoothing image restoration procedure based on a growing region type algorithm; the second step uses parameters estimated during the first phase to produce an estimation of the boundaries of the image segmentation. The last step of the algorithm groups the different areas obtained during the second phase by minimising a penalized empirical squared loss criterion. The segmentation algorithm is then applied to simulated nuclear magnetic resonance data.

Remerciements

Ce travail a été effectué dans le cadre du projet « Méthode de Classification et Imagerie par Ondelettes : Aide au diagnostic pour les tumeurs cérébrales » financé par la région Rhones-Alpes et en collaboration avec l'INSERM Unité 594, et le service IRM du CHU de Grenoble.

1. Introduction

Les technologies récentes utilisant la Résonance Magnétique Nucléaire (RMN) pour étudier la composition tissulaire du cerveau (Imagerie par Résonance Magnétique) donnent accès à une quantité toujours plus grande d'information. Cet enrichissement des données, s'il permet d'espérer des caractérisations tissulaires plus fines, n'en rend pas moins l'analyse plus ardue. Aussi, l'imagerie médicale, et plus généralement l'imagerie hyper-spectrale, nécessitent aujourd'hui des méthodes statistiques nouvelles, adaptées à la grande dimension des données, et c'est dans ce cadre que se situe ce travail.

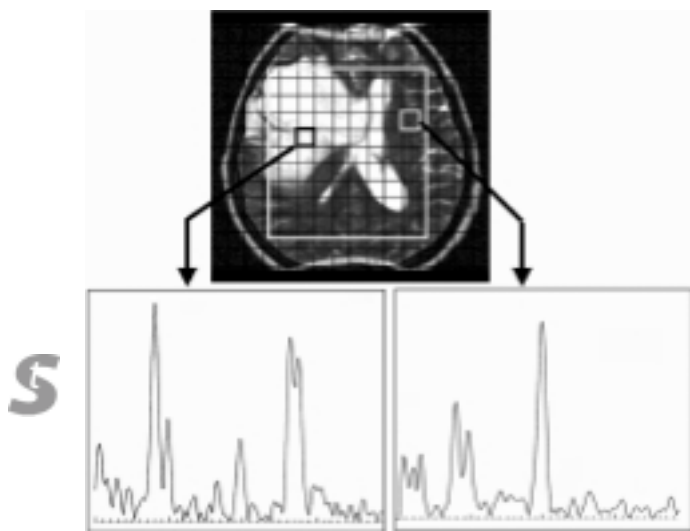


Figure 1. Images hyperspectrales d'une coupe du cerveau.

L'imagerie spectroscopique (voir [6] pour une introduction à l'imagerie par résonance magnétique en général) est la forme d'imagerie médicale qui permet d'obtenir l'acquisition simultanée de l'ensemble des spectres localisés dans un plan de coupe du cerveau. Chaque volume élémentaire de tissu (voxel) est alors caractérisé par un spectre, c'est-à-dire la réponse en fréquence des différentes substances chimiques qui le compose (voir Figure 1). Ces spectres sont issus de la réaction magnétique des molécules d'eau à une impulsion de radio fréquence, et permettent par exemple d'associer au spectre de chaque voxel d'une image spectroscopique de patients un type histopathologique de tumeur. Les études existantes en traitement de signaux issus d'examen IRM n'utilisent pas souvent la caractérisation spectroscopique des tissus, et même lorsque des études de ce type sont effectuées, elles ne s'attachent qu'au spectre correspondant à un voxel dont on sait à l'avance qu'il est issu d'une tumeur dont on veut connaître les propriétés.

Depuis plusieurs années un certain nombre de travaux basés sur le traitement d'images ont été réalisés pour essayer de mettre en évidence des différences entre les spectres de tumeurs de types

et de grades différents (voir par exemple Szabo De Edeleneyi ([22]) ou le résumé des techniques existantes fait par Hagberg ([10])). Le point commun entre tous ces travaux est que les données enregistrées ne sont pas traitées comme des données fonctionnelles mais se limitent plutôt à un choix *a priori* de certains attributs caractéristiques des spectres observés. Ces résumés vectoriels sont ensuite utilisés pour discriminer ou prédire le type histopathologique des tissus avec des techniques classiques de reconnaissance de forme ou d'analyse statistique multidimensionnelle. Les indicateurs (attributs) le plus souvent utilisés sont des bandes spectrales dont on sait qu'elle apportent une information pertinente. Il nous a semblé donc naturel et désirable de rechercher à utiliser des méthodes de classification et de segmentation exploitant toute l'information disponible, à savoir la totalité des spectres observés dans l'image spectroscopique.

L'objectif de ce travail est, grâce à une technique statistique de réduction de dimension, de proposer un algorithme de segmentation ne nécessitant pas de réduction *a priori* de la dimension des spectres et particulièrement adapté à la segmentation d'images hyper-spectrales fortement bruitées. La complexité des enregistrements analysés justifie en partie l'approche simplificatrice consistant à supposer que l'image est composée de zones homogènes, ce que nous ferons dans le reste de ce travail. Ce texte est articulé en quatre parties. La première consiste en une description des hypothèses mathématiques et sert d'introduction pour la deuxième dédiée à un algorithme de débruitage d'images hyper-spectrales inspiré de l'algorithme AWS proposé par Polzehl et Spokoiny ([16]) pour le débruitage d'images. Cet algorithme ne produit pas de segmentation et c'est pour cela que dans la troisième partie nous montrons comment cet algorithme peut fournir un estimateur des contours de régions de l'image qui débouche sur une méthode de segmentation. La dernière partie est consacrée à des simulations, et notre démarche est comparée avec celle de Whitcher *et al.* [24]. Ces auteurs s'intéressent à des techniques de classification non supervisée fondées sur la décomposition en ondelettes des spectres issus d'une image hyper-spectrale.

2. Généralités et notations

Nous supposons que l'image étudiée est homogène par régions, *i.e.* que c'est une fonction $f : [0, 1]^2 \rightarrow L^2[0, 1]$ (donc à valeur dans un espace de fonctions) pour laquelle il existe M régions distinctes $A_m, m = 1 \dots M$, telles que :

$$\forall x \in [0, 1]^2 \quad f(x) = \sum_{m=1}^M a_m \mathbb{1}_{A_m}(x). \quad (1)$$

Dans l'expression (1), a_m désigne un élément de $L^2[0,1]$ et $\mathbb{1}_{A_m}(x)$ est l'indicatrice de l'ensemble A_m . Le nombre M de régions, les régions A_m , et les valeurs a_m de f sur ces régions sont inconnus. On supposera, pour des raisons évidentes d'identifiabilité et sans perte de généralité, que les a_m sont tous distincts. La fonction f est observée sur une grille régulière du plan $X_i = \left(\frac{i_1}{p}, \frac{i_2}{p}\right)$ avec $i = (i_1, i_2) \in \{1, \dots, p\}^2$ dans un bruit blanc gaussien η_i multiplié par un facteur d'échelle σ_i :

$$S_i = f(X_i) + \sigma_i \eta_i ; i \in \{1, \dots, p\}^2 \quad (2)$$

L'indice i désignera un indice double identifiant de manière unique la position d'un pixel (c'est-à-dire, en terme d'imagerie médicale un voxel), et pour deux pixels différents X_i et X_j , les bruits η_i et η_j ne seront pas corrélés.

Le problème que l'on se pose dans la suite est d'une part d'estimer la fonction f et d'autre part d'identifier le nombre et les régions sur lesquelles f est homogène (*i.e.* les zones sur lesquelles f est constante).

Nous rappelons que pour une image composée de niveaux de gris, f est à valeur dans le corps des nombres réels \mathbb{R} , pour une image RVB, f est à valeur dans \mathbb{R}^3 . Dans l'idéal, un spectre bruité S_i (une variable aléatoire à valeur dans $L^2[0,1]$) est observé pour chaque volume élémentaire du plan de coupe du cerveau considéré. En pratique les courbes aléatoires S_i ne sont connues qu'en un ensemble de points discret. Elles devront donc être approchées par certaines fonctions définies sur tout $[0,1]$, et cette approximation sera dans notre cas réalisée en développant les S_i dans une base d'ondelettes, les coefficients de la décomposition étant estimés à partir des données discrétisées. Pour notre application, nous utiliserons une transformée en ondelettes périodiques (voir par exemple [14]) et nous considérerons donc observer en chaque pixel i , $N = 2^J$ coefficients d'ondelette bruités, notés $\theta_\nu(X_i)$:

$$\begin{aligned} (P_{E_N}(S_i))_\nu &= Y_\nu(X_i) \\ &= \theta_\nu(X_i) + \sigma(X_i)\epsilon_\nu(X_i) \quad \nu \in \{1 \dots N\} \end{aligned} \quad (3)$$

où P_{E_N} désigne la projection du spectre S_i sur l'espace d'approximation d'échelle J . Grâce à l'orthogonalité de la transformée en ondelette, les $\epsilon_\nu(X_i)$ constituent encore un bruit blanc gaussien. L'indice $\nu = (j, k)$ est un couple composé d'un indice d'échelle (noté $|\nu| = j$) et d'un indice de position k .

Il est connu que pour une grande classe de fonctions, la représentation dans le domaine des ondelettes est creuse, autrement dit la plupart des fonctions usuelles sont caractérisées par un petit nombre de coefficients non nuls dans la base d'ondelette et ceci sera exploité dans la suite. Toutes nos estimations seront faites sur les coefficients d'ondelette. Il est alors possible d'estimer $f(X_i)$ par une transformée inverse des coefficients d'ondelette estimés.

2.1. Synthèse de l'algorithme

Avant de présenter en détail les trois phases différentes qui vont nous permettre d'aboutir à la segmentation d'une image hyper-spectrale, nous en donnons un aperçu rapide.

- **Phase 1** : Estimation de $w_{ij} = 1_{f(i)=f(j)}$ par \hat{w}_{ij} grâce à la combinaison d'un procédé de réduction de dimension et de l'algorithme AWS de Pozhel et Spokoyny ([16]) (section 3)
- **Phase 2** : Détection des frontières par un système de votes utilisant l'estimation des poids fournie par la phase 1. (section 4)
- **Phase 3** : Regroupement des régions obtenues lors de la phase 2 par une méthode de minimisation de l'erreur quadratique empirique pénalisée (section 5).

3. AWS : Algorithme de débruitage et d'estimation des poids

L'algorithme que nous allons décrire et utiliser est inspiré dans ses grandes lignes par l'algorithme AWS de Pozhel et Spokoyny ([16]). Ces derniers ont appliqué leur algorithme à des données vectorielles ([17]) (des données modélisées par des fonctions du type (1) pour lesquelles a_m est un vecteur réel multidimensionnel). Leur application est donc radicalement différente de celle que l'on considère ici car dans leur démarche les vecteurs a_m ne tiennent pas compte de l'aspect fonctionnel des données. En d'autre terme, l'originalité de la méthode décrite ici réside dans la combinaison d'une méthode de réduction de dimension (qui correspond à utiliser le caractère fonctionnel des données) et de l'algorithme AWS.

L'algorithme est itératif. Étant donné une estimation $(\hat{\theta}_i^{k-1})_i$ des vecteurs des coefficients d'ondelette de f obtenue à l'étape $k-1$, pour chaque pixel i , la $k^{\text{ème}}$ itération est composée de deux étapes :

- **étape 1** : Sélectionner le sous ensemble V_i^{k+} des pixels de V_i^k voisins de i qui appartiennent à la même zone que i grâce à $(\hat{\theta}_i^{k-1})_i$ (la méthode de sélection correspondante est décrite au paragraphe suivant).
- **étape 2** : Estimer la valeur de θ_i le vecteur des coefficients d'ondelette au pixel i , grâce à une moyenne des observations sélectionnées à l'étape 1.

La suite de voisinages $(V_i^k)_k$ des pixels susceptibles d'appartenir à la même zone que i est destinée à croître et doit être choisie par l'utilisateur de l'algorithme. L'important est que dans les premières étapes de l'algorithme le voisinage soit suffisamment petit pour permettre une réduction du bruit sans trop d'erreurs de sélection.

Après une description de la méthode de sélection, l'algorithme sera détaillé de manière plus formelle et plus précise dans la section 3.3.

3.1. Méthode de sélection

Pour chaque pixel i , l'algorithme que nous allons décrire estime progressivement l'unique zone $A_{m(i)}$ à laquelle i appartient. Il sélectionne les pixels d'un voisinage V_i de i qui sont vraisemblablement dans $A_{m(i)}$. Cette sélection est formalisée grâce à un ensemble de tests des hypothèses suivantes :

$$\forall j \in V_i \quad H_{0j} : j \in A_{m(i)} \quad \text{Vs} \quad H_{1j} : j \notin A_{m(i)}. \quad (4)$$

L'estimateur de $V_i^+ = V_i \cap A_{m(i)}$ est tout naturellement

$$\widehat{V}_i^+ = \{j \in V_i \mid H_{0j} \text{ accepté}\}. \quad (5)$$

Il s'agit donc à présent d'utiliser un test pour tester les hypothèses de (4), ce qui revient à tester si les fonctions de $L^2[0,1]$ $f(X_i)$ et $f(X_j)$ sont identiques ou pas pour $j \in V_i$, i.e :

$$\forall j \in V_i \quad H_{0j} : \|f(X_j) - f(X_i)\|_{L^2[0,1]} = 0 \quad \text{Vs} \quad H_{1j} : \|f(X_j) - f(X_i)\|_{L^2[0,1]} \neq 0. \quad (6)$$

Le test que nous utilisons et que nous allons présenter a été introduit dans ([18]) et ([8]).

Remarquons que l'orthogonalité de la base d'ondelettes périodiques de $L^2[0,1]$ permet de réécrire (6) de la manière suivante :

$$\forall j \in V_i \quad H_{0j} : \|(\theta_\nu(X_j))_{\nu \in \mathbb{N}} - (\theta_\nu(X_i))_{\nu \in \mathbb{N}}\|_{l^2} = 0 \quad \text{Vs} \quad H_{1j} : \|(\theta_\nu(X_j))_{\nu \in \mathbb{N}} - (\theta_\nu(X_i))_{\nu \in \mathbb{N}}\|_{l^2} \neq 0. \quad (7)$$

Au vu de l'approximation de f par sa projection dans E_N nous confondrons les hypothèses de (7) avec :

$$\forall j \in V_i \quad H_{0j} : \|(\theta_\nu(X_j))_{\nu \in \{1 \dots N\}} - (\theta_\nu(X_i))_{\nu \in \{1 \dots N\}}\|_2 = 0 \quad \text{Vs} \quad H_{1j} : \|(\theta_\nu(X_j))_{\nu \in \{1 \dots N\}} - (\theta_\nu(X_i))_{\nu \in \{1 \dots N\}}\|_2 \neq 0 \quad (8)$$

ce qui revient finalement à tester la nullité de la moyenne d'un certain nombre de vecteurs gaussiens de dimension N , dont la dimension est généralement grande.

3.2. Test d'hypothèses fonctionnelles : réduction de dimension

Soit μ un vecteur de \mathbf{R}^N composé des $N = 2^J$ premiers coefficients d'ondelette d'une fonction g . Nous supposons que ce vecteur est observé dans un bruit gaussien à composantes indépendantes et identiquement distribuées de variance σ et de moyenne nulle :

$$Z_\nu = \mu_\nu + \sigma \epsilon_\nu \quad \nu = 1 \dots N.$$

Nous allons décrire une procédure pour tester au vu de ces observations l'hypothèse :

$$H_0 : \|\mu\|_2 = 0 \quad \text{Vs} \quad H_1 : \|\mu\|_2 > \rho_N. \quad (8)$$

Nous supposons σ connu. En pratique, nous avons utilisé l'estimation définie par la médiane des valeurs absolues des écarts à la médiane divisée par 0,6745, appliquée aux coefficients d'ondelettes à l'échelle la plus fine. Ceci produit un estimateur robuste de σ (voir par exemple [14]).

Il est nécessaire lorsque l'on veut tester la nullité de μ dans (8) que l'hypothèse alternative soit suffisamment séparée de l'hypothèse nulle (voir par exemple [11]) et ce afin que le test n'ait pas une puissance trop faible. La valeur ρ_N est un seuil de séparabilité entre l'hypothèse nulle et l'alternative garantissant une puissance satisfaisante (voir [11]).

Nous rappelons que pour une vaste gamme de fonctions g , μ est creux, c'est-à-dire ne contient que peu de coefficients non nuls. Cela explique que la construction de la statistique de test passe par l'estimation aux niveaux j tels que $j_s \leq j \leq \log_2(N)$ des ensembles $I_j \subset \{\nu \text{ tels que } |\nu| = j\}$ des indices des coefficients d'ondelette non nuls. Les coefficients d'échelle plus grossière sont conservés tels quels :

$$\widehat{I}_j = \begin{cases} \left\{ \nu \text{ tels que } |\nu| = j \text{ et } |Z_\nu| > \sigma 4\sqrt{8\log(2^{j-j_s})} \right\} \\ j_s \leq j \leq \log_2(N) = J \\ \{\nu \text{ tels que } |\nu| = j\} \quad 0 \leq j \leq j_s \end{cases}. \quad (9)$$

L'énergie de ces coefficients est donnée par :

$$\mathcal{E}(Z)^2 = \sum_{\nu \in \cup_{j=1}^J \widehat{I}_j} Z_\nu^2. \quad (10)$$

Si cette énergie est trop importante on choisit de rejeter H_0 . La construction de la statistique de test correspondante ainsi que le choix du seuil à partir duquel on décide de rejeter H_0 nécessite le calcul de $E[\mathcal{E}(Z)^2]$ et de $\text{Var}(\mathcal{E}(Z)^2)$ sous H_0 (notés $E_0[\mathcal{E}(Z)^2]$ et $\text{Var}_0(\mathcal{E}(Z)^2)$). Un calcul exact de ces expressions est donné dans [1]. La statistique de test est obtenue en centrant et en normalisant l'énergie :

$$T(Z) = \frac{\mathcal{E}(Z)^2 - E_0[\mathcal{E}(Z)^2]}{\sqrt{\text{Var}_0(\mathcal{E}(Z)^2)}}. \quad (11)$$

Sous H_0 , cette statistique suit asymptotiquement une loi normale centrée réduite (voir [18]). La région de rejet correspondante est :

$$R_\lambda^H = \{T > \lambda\} \quad (12)$$

où λ peut être choisi pour assurer une probabilité de fausse alarme plus petite que α : $\lambda = z_{1-\alpha}$ (z_α est le quantile d'ordre α de la loi normale centrée réduite) .

Remarque. Le choix du seuil $4\sqrt{8\log(2^{j-j_s})}$ dans (9) est discuté par Spokoiny ([18]), celui-ci montre aussi les bonnes pro-

priétés du test lorsque g appartient à un espace de Besov. La recherche du paramètre j_s (le nombre de niveaux qui ne sont pas seuillés) doit en théorie être guidé par la connaissance a priori de la régularité de la fonction. Spokoiny ([18]) propose une méthode de sélection de j_s par un algorithme adaptatif. Cependant, dans la pratique avec des tailles de trajectoires de 256 et donc pour notre application en particulier, le choix $j_s = 3$ réalise un bon compromis.

3.3. L'algorithme AWS

À partir de maintenant et dans la suite nous noterons $\theta_i = (\theta_\nu(X_i))_{\nu \in \{1 \dots N\}}$, et $Y_i = (Y_\nu(X_i))_{\nu \in \{1 \dots N\}}$, le vecteur des coefficients d'ondelette à estimer au pixel i et les observations correspondantes introduites en (3).

Pour chaque pixel i on définit une suite croissante de voisinages $(V_i^k)_k$. Le plus petit d'entre eux, point de départ de l'algorithme, est réduit à un pixel: $V_i^0 = \{i\}$. Il est évident que $V_i^{0+} = A_{m(i)} \cap V_i^0 = \{i\}$. Étant donné une estimation \widehat{V}_i^{k+} de l'ensemble $V_i^{k+} = A_{m(i)} \cap V_i^k$ des pixels du voisinage de i qui sont dans la même zone que i , un bon estimateur de θ_i est simplement obtenu par la moyenne des observations de \widehat{V}_i^{k+} :

$$\hat{\theta}_i^k = \frac{1}{|\widehat{V}_i^{k+}|} \sum_{j \in \widehat{V}_i^{k+}} Y_j,$$

$|\widehat{V}_i^{k+}|$ désignant le cardinal de \widehat{V}_i^{k+} .

L'algorithme consiste à itérer l'étape de sélection suivante de $k = 1$ à k_{\max} :

Sélection: Pour chaque pixel i , la sélection se décompose en 2 étapes:

- Pour chaque pixel j de V_i^k , poser:

$$\Delta_{ij}^k = \hat{\theta}_i^{k-1} - \hat{\theta}_j^{k-1}. \quad (12)$$

Appliquer la technique de réduction de dimension décrite dans la sous-section 3.2:

$\forall j \in V_i^k, T_{ij}^k = T(\Delta_{ij}^k)$, T étant donné par (11).

- D'après (5), estimer V_i^{k+} par l'ensemble des j pour lesquels H_{0j} est accepté:

$$\widehat{V}_i^{k+} = \{j \in V_i^k \text{ tels que } T_{ij}^k \leq \lambda_i^k\}.$$

L'estimateur final de θ_i est donc un estimateur à noyau:

$$\hat{\theta}_i^{k_{\max}} = \frac{1}{|\widehat{V}_i^{k_{\max}+}|} \sum_{j \in \widehat{V}_i^{k_{\max}+}} Y_j = \sum_j \frac{\hat{w}_{ij}^{k_{\max}}}{|\widehat{V}_i^{k_{\max}+}|} Y_j$$

$$\text{avec } \hat{w}_{ij}^{k_{\max}} = 1_{\widehat{V}_i^{k_{\max}+}}. \quad (13)$$

Remarque 3.2: Si à l'étape $k - 1$ un certain nombre de sélections à tort ont été faites, Δ_{ij}^k a une grande probabilité de ne pas être d'espérance nulle sous H_{0j} . Ceci peut être la cause de nouvelles erreurs et on peut craindre que ce biais ne se propage d'étape en étape. D'autre part, si on cherche à diminuer λ (pour éviter les erreurs), peu de pixels seront sélectionnés; la variabilité de la statistique de test sera donc trop importante et induira une trop forte probabilité d'erreur: faux positifs (sélection à tort) et faux négatifs (non sélection à tort). Tout le travail pratique et théorique consiste donc à trouver la bonne valeur de λ et à vérifier qu'asymptotiquement pour une certaine classe de valeurs « admissibles » de λ , le biais d'estimation n'explose pas et la variabilité de l'estimation décroît vers 0. Le problème est qu'il est difficile d'obtenir des résultats théoriques sur la convergence de cette procédure et ce à cause de (13) et de la manière dont sont construits les poids. Cette construction établit en effet une structure de dépendance très complexe entre les poids d'une part et entre les poids et les observations d'autre part.

D'après cette remarque, le choix de λ_i^k est crucial, il conditionne le bon comportement de l'algorithme. Cependant par soucis de clarté, les deux méthodes de sélection, celle de Pozhel et Spokoiny et une autre procédure que nous avons envisagée, sont exposées en annexe.

Finalement, l'algorithme AWS est une méthode d'estimation de f . L'estimateur final de f au pixel i étant obtenu par une moyenne des observations dans une fenêtre sélectionnée \widehat{V}_i^{k+} , elle s'apparente à une méthode d'estimation à noyau non linéaire. Elle ne produit pas de segmentation de l'image, cependant les \hat{w}_{ij} sont une estimation des poids qui seront utilisés dans la section suivante pour la phase de segmentation:

$$w_{ij} = 1_{f(i)=f(j)} = \begin{cases} 1 & \text{si } i \text{ et } j \text{ sont dans la même zone,} \\ 0 & \text{sinon.} \end{cases}$$

L'utilisation de ces poids fournis par l'algorithme AWS a déjà été envisagée par Philippe *et al.* [15] pour détecter des ruptures dans un signal unidimensionnel mais ne l'a jamais été, à notre connaissance, pour la segmentation.

Remarque: Si à chaque étape k de l'algorithme en élargissant V_i^k on cherche à faire grandir l'ensemble des pixels j dont on pense qu'ils sont dans la même zone que i , cette méthode n'est pas une variante des méthodes de segmentation par croissance de région telles qu'elles sont décrites dans ([25]). Les régions ne sont pas en « compétition »: pour chaque pixel i un ensemble de poids \hat{w}_{ij} est obtenu, mais il n'est pas exclu que $\hat{w}_{ij} = 1$, $\hat{w}_{ik} = 1$ et que $\hat{w}_{jk} = 0$.

4. Segmentation par estimation des frontières

Nous allons présenter ici notre méthode de segmentation. Cette méthode repose sur l'extraction des contours par un système de votes. L'originalité de la méthode réside dans l'exploitation statistique (basée sur le système de votes) des poids de AWS pour la segmentation. Une méthode simple est ensuite utilisée pour obtenir les régions de l'image délimitées par les contours obtenus.

4.1. Estimation de frontières par une méthode de vote

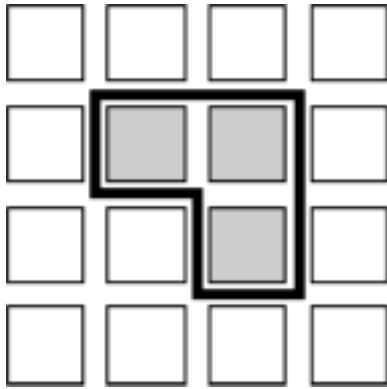


Figure 2. Description interpixelaire d'une frontière.

Dans le but d'obtenir une description « inter-pixelaire » des frontières (voir figure 2), nous allons chercher à construire une statistique qui indiquera de manière fiable si un segment séparant deux pixels adjacents constitue une frontière ou non. À partir de maintenant et dans toute la suite, \mathcal{S} désignera l'ensemble des segments séparant deux pixels adjacents dans l'image. Pour une image carrée, composée de p^2 pixels, le cardinal de \mathcal{S} est $2p(p-1)$. À un élément u de \mathcal{S} sont associés les deux pixels qu'il sépare: $u_g \in \{1, \dots, p\}^2$ et $u_d \in \{1, \dots, p\}^2$. Un tel segment est une frontière entre deux régions si et seulement si $f(u_g) \neq f(u_d)$. Il est donc possible, comme cela a été fait par Philippe *et al.*[15] pour détecter des ruptures dans un signal, de décider que le segment u sépare les deux pixels adjacents u_g et u_d si $\hat{w}_{u_d u_g} = 0$.

Néanmoins, cette estimation des segments de frontière est trop variable. Pour réduire cette variabilité, il est possible, d'utiliser plus d'information. En effet, un segment u de \mathcal{S} est un morceau de frontière si et seulement si

$$\forall i \in \{1, \dots, p\}^2 \quad 1_{f(i) \neq f(u_g)} \neq 1_{f(i) \neq f(u_d)}. \quad (15)$$

Cette caractérisation d'un segment de frontière utilise les observations associées à tous les pixels. Nous appellerons vote du

pixel i pour la frontière u entre deux pixels voisins u_g et u_d , la variable aléatoire :

$$Y_i(u) = 1_{\hat{w}_{i u_g}^k \neq \hat{w}_{i u_d}^k}$$

Plus un morceau de frontière totalise de vote plus il est probable qu'il soit en effet une frontière. La totalité des votes $Y_i(u)$ comptabilisés pour des votant i contenus dans un voisinage $Vois(u)$ sera noté $M(u, Vois(u))$. À un ensemble de segment Γ on peut donc associer la statistique qui comptabilise le nombre normalisé de votes pour les segments de Γ :

$$M(\Gamma) = \frac{1}{\sqrt{n(\Gamma)}} \sum_{u \in \Gamma} M(u, Vois(u))$$

$$M(u, Vois(u)) = \sum_{i \in Vois(u)} Y_i(u)$$

$$n(\Gamma) = \sum_{u \in \Gamma} |Vois(u)|$$

En admettant que tous les poids sont indépendants (ce qui n'est qu'une approximation) l'inégalité de Hoeffding permet d'obtenir la proposition suivante :

Proposition 4.1. : Si un ensemble de segments inter-pixelaire Γ n'est composé d'aucune frontière, alors

$$P_0 (M(\Gamma) - E_0[M(\Gamma)] > u_\delta) \leq \delta.$$

pour un seuil $u_\delta = (\frac{1}{2} \log(1/\delta))^{1/2}$ (l'espérance $E_0[\]$ étant prise sous l'hypothèse qu'il n'y a pas de frontières dans Γ).

Le calcul de $E_0[M(\Gamma)]$ ainsi qu'une méthode pour l'estimer est donnée en annexe. Cette proposition donne une valeur de seuil à partir de laquelle un ensemble de segments Γ est considéré comme une frontière. Nous déciderons qu'un ensemble de segments Γ cohérents (voir sous section suivante) correspondra à une frontière si

$$M(\Gamma) - E_0[\widehat{M}(\Gamma)] > u_\delta, \quad (16)$$

où $E_0[\widehat{M}(\Gamma)]$ est une estimation $E_0[M(\Gamma)]$ donnée en annexe. Dans les applications, nous avons utilisé $\delta = 0,01$.

4.2. Partition de l'image en zones homogènes

Une zone connexe A_i qui n'est pas en contact avec les bords de l'image est délimitée par une frontière $\partial A_i = U_i \partial A_i^l$ où chaque morceau de frontière ∂A_i^l est constitué de segments de \mathcal{S} et forme un cycle sans boucle de \mathcal{S} . Si la zone connexe A_i est en contact avec le bord de l'image, alors sa frontière est composée de $\partial A_i = U_i \partial A_i^l$ où chaque ∂A_i^l est constituée de segments de \mathcal{S} et forme soit un cycle sans boucle de \mathcal{S} , soit relie un point du bord de l'image à un autre.

Nous allons donc chercher dans une zone \mathcal{P} de l'image un cycle ou une suite de segments reliant un bord de \mathcal{P} à un autre bord de \mathcal{P} . La procédure correspondante est appelée recherche-une-frontière(\mathcal{P}). Cette procédure retourne une frontière Γ dont la somme des poids est aussi importante que possible. Trois cas sont alors envisageables :

- Soit Γ vérifie (16) et divise \mathcal{P} en deux régions. Dans ce cas, la frontière est acceptée et la procédure est rappelée pour ces deux régions de l'image.

- Soit Γ vérifie (16) et ne divise pas \mathcal{P} en deux régions. Dans ce cas, la frontière est acceptée et la procédure est de nouveau appliquée à l'ensemble \mathcal{P} et en considérant Γ comme une frontière.

- Si Γ ne vérifie pas (16) alors la procédure s'arrête, et Γ n'est pas considéré comme une frontière.

En appliquant cette procédure récursive à l'image toute entière, on produit donc une segmentation de cette dernière : on divise l'image en B_1, \dots, B_n ensembles disjoints de pixels.

Dans notre application, nous avons mis en place une procédure recherche-une-frontière(\mathcal{P}) qui choisit le segment u tel que $M^k(u)$ soit maximum dans \mathcal{P} . Ensuite, on se déplace vers le segment v en contact avec u qui maximise $M^k(v)$. Cette démarche est répétée jusqu'à l'obtention d'un cycle sans boucle ou d'une suite de segments reliant un bord de \mathcal{P} à un autre bord de \mathcal{P} ou d'une suite de segments se recoupant.

5. Regroupement des zones par minimisation de l'erreur quadratique empirique pénalisée

L'étape précédente a permis d'obtenir B_1, \dots, B_n , n ensembles disjoints de pixels (que l'on assimile à des indices doubles) sur lesquels f est supposée être constante. Notre but est de chercher à regrouper les composantes non nécessairement connexes de l'image que la procédure précédente n'a pas regroupé. Soit \mathcal{P} l'ensemble des partitions de l'image que l'on peut obtenir en fusionnant certains des $(B_i)_{i=1, \dots, n}$. Si p est un élément d'une partition $P \in \mathcal{P}$, $Ave(p)$ désignera la moyenne des observations de p . On cherchera à garder la partition \hat{P} qui rend petit le risque quadratique :

$$\mathcal{R}_P = \sum_{p \in \mathcal{P}} \sum_{i \in p} \frac{\|Y_i - Ave(p)\|_{2,n}^2}{\sigma(X_i)^2},$$

où $\|x\|_{2,n} = \frac{1}{n} \sum_{i=1}^n x_i^2$ et $\sigma(X_i)$ est calculé selon la procédure décrite dans la section 3.2.

Avec ce critère, il est clair que c'est la partition qui possède le plus de régions qui va être favorisée. Aussi est-il nécessaire

d'appliquer une pénalité, fonction du nombre de régions. Cette pénalité est le plus souvent assez complexe, mais une pénalité proportionnelle au nombre d'éléments de la partition (autrement dit à la dimension du modèle, qui correspond à supposer que notre fonction f est constante sur chaque partie de la partition). La pénalité $|P|\eta$ permet d'obtenir de bons résultats autant d'un point de vue théorique que pratique. La segmentation finale choisie est donc

$$\hat{P} = \text{Argmin}_{P \in \mathcal{P}} \{\mathcal{R}_P + |P|\eta\}.$$

Le paramètre η peut être choisi par un critère de validation croisé, mais un autre choix raisonnable est également $\eta = 2 \log(p^2)$ (voir par exemple [7] ou [8] pour une justification théorique).

D'un point de vue algorithmique, nous utiliserons le principe de CART ([5]). Etant donnée une partition $P \in \mathcal{P}$ composée de n_P régions, on choisit de regrouper les deux éléments p_1 et p_2 de cette partition qui minimisent

$$\mathcal{R}_{\{p_1 \cup p_2\}} = \sum_{i \in p_1 \cup p_2} \frac{\|Y_i - Ave(p_1 \cup p_2)\|_2^2}{\sigma(X_i)^2}.$$

Si la nouvelle partition \tilde{P} induite par un tel regroupement vérifie :

$$\mathcal{R}_{\tilde{P}} - \mathcal{R}_P < 2 \log(p^2),$$

c'est-à-dire si

$$\mathcal{R}_{\{p_1 \cup p_2\}} - (\mathcal{R}_{\{p_1\}} + \mathcal{R}_{\{p_2\}}) < 2 \log(p^2),$$

alors le regroupement est accepté. On cherche alors un nouveau regroupement. Sinon il est rejeté et l'algorithme s'arrête. L'astuce algorithmique consiste à calculer $\mathcal{R}_{\{p_1 \cup p_2\}}$ grâce à la formule de Huygens :

$$\begin{aligned} \mathcal{R}_{\{p_1 \cup p_2\}} &= \mathcal{R}_{\{p_1\}} + \mathcal{R}_{\{p_2\}} + |p_1|(Ave(p_1) \\ &\quad - Ave(p_1 \cup p_2))^2 + |p_2|(Ave(p_2) - Ave(p_1 \cup p_2))^2, \end{aligned}$$

$$Ave(p_1 \cup p_2) = \frac{1}{|p_1 \cup p_2|} (|p_1|Ave(p_1) + |p_2|Ave(p_2)).$$

6. Application

Nous présentons maintenant les résultats obtenus par notre algorithme pour une expérience proposée par Witcher *et al.* [24]. L'image hyper-spectrale est une image 64×64 , pour laquelle en chaque pixel, un spectre discrétisé sur 128 points est observé dans un bruit blanc gaussien. L'image est divisée en 12 régions actives (celles sur lesquelles le signal est non nul) réparties en trois colonnes de quatre carrés de 16×16 pixels. Sur chacun des pixels de ces carrés le signal est de la forme

$f(t) = a + b(e^{-t/T_{out}} - e^{-t/T_{in}})$ (voir Figure 3). Le lecteur pourra consulter [24] pour une explication plus détaillée, ces signaux temporels sont une version simplifiée de ce que les équations théoriques de Bloch (équation différentielle de conservation du moment magnétique) permettent d'obtenir lors d'une séquence d'acquisition RARE (ces signaux ne sont donc pas observés dans le domaine fréquentiel). Les paramètres T_{out} et T_{in} changent d'une colonne à l'autre mais restent les mêmes d'une ligne à l'autre. Les amplitudes maximales de $f(t)$ varient d'une ligne à l'autre et restent les mêmes d'une colonne à l'autre afin que le rapport contraste sur bruit (CNR) dans chaque ligne soit respectivement de 6,4,2 et 1; ceci détermine les constantes a et b . La quatrième colonne de l'image ne contient que du bruit. La forme spatiale des régions actives est rendue circulaire de diamètre 8 par la convolution de chaque carré avec une gaussienne dont le maximum coïncide avec le centre du carré. Cette convolution est faite avant que le bruit gaussien ne soit ajouté et a donc pour effet de faire varier localement le CNR sans changer le bruit.

Les résultats obtenus sont présentés à la Figure 6. Tous les cercles sont différenciés sauf les cercles de la première colonne et de la troisième colonne pour la première ligne (les deux spectres qui se ressemblent le plus pour le rapport contraste sur bruit le plus faible). Il reste aussi deux pixels isolés d'affectation erronée. Enfin les cercles sont quelque peu déformés. Les résultats obtenus sont radicalement différents de ceux obtenus par Witcher *et al.*[24]. En effet, mis à part quelques pixels affectés à des zones isolées après l'étape de détection de frontières, nos zones sont régulières dans le sens où elles n'ont que peu ou pas de trous correspondant à des erreurs. Ceci résulte directement de la structure de notre procédure combinant une méthode de lissage par noyau (non linéaire) à une méthode de segmentation. Ceci a pour effet d'une part de produire une segmentation dans

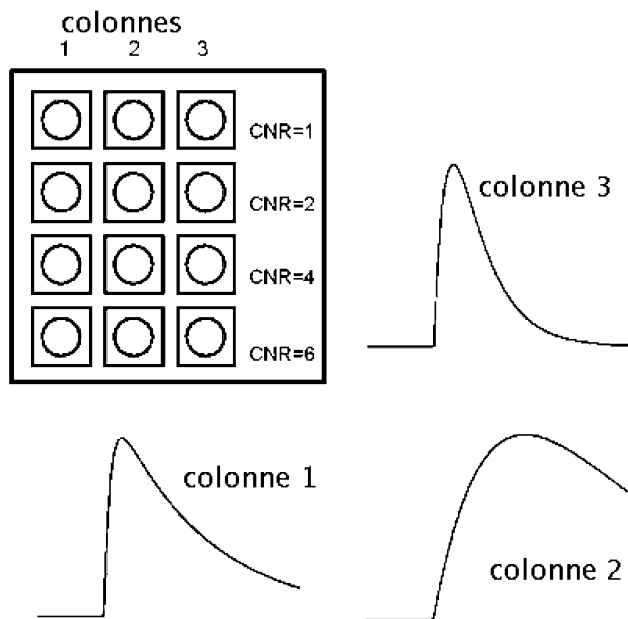


Figure 3. Construction de l'image hyper-spectrale.

laquelle les distances dans l'image ont leur rôle à jouer et d'autre part de réduire assez significativement le bruit pour pouvoir différencier certaines zones regroupées à tort dans la démarche de Witcher *et al.* Notre algorithme tire pleinement parti de la taille des zones homogènes et de la régularité des frontières.

Nous tenons aussi à remarquer que l'image utilisée n'est pas vraiment constante par régions. En effet, chaque carré étant convolé avec une gaussienne d'intensité maximale au centre du carré, les cercles détectés ne sont pas séparés de l'image par une frontière matérialisée par une rupture, mais corres-



Figure 4. Image hyper-spectrale construite colonne 3 ligne 3 et 2.



Figure 5. Image hyper-spectrale construite colonne 3 ligne 1 (CNR=1)

pondent en fait à une zone d'inflexion de la gaussienne utilisée lors de la convolution. Ceci nous permet de conclure à une certaine robustesse de l'algorithme utilisé.

7. Conclusion et perspectives

Nous avons présenté un algorithme de segmentation d'images hyper-spectrales. Cet algorithme tire parti de l'homogénéité de certaines zones et de la régularité des frontières les délimitant pour séparer de manière convaincante les différentes régions. Notons que la phase d'estimation donne lieu à des problèmes théoriques difficiles liés à la dépendance. La partie algorithmique de recherche de frontière de poids maximum mériterait d'être améliorée en utilisant par exemple des algorithmes du

type minimum ratio weight cycle (voir Jermyn [12]) et des algorithmes plus rapides de formation de zones à partir des frontières (voir par exemple [9]). Cela fera l'objet d'un travail ultérieur.

A. Choix des paramètres de AWS

Nous allons décrire dans les sous-sections suivantes deux manières de déterminer les paramètres $(\lambda_i^k)_{i,k}$ introduits dans la sous section 3.3.

A.1. Contrôle des faux positifs

Soit i un pixel fixé, nous cherchons à décrire les quantités liées aux tests faits à l'étape k dans V_i^k . Nous rappelons qu'un rejet de H_{0j} à tort correspond à une fausse détection de frontière (fausse alarme) c'est-à-dire un faux positif pour le pixel $j \in V_i^k$. On remarque que les espérances des nombres de vrais positifs et de faux positifs sont respectivement égales au nombre moyen de pixels $j \in V_i^k$ sélectionnés à raison :

$$\sum_{j \in V_i^k, H_{0j} \text{Vrai}} P(H_{0j} \text{ accepté}) = E [|\widehat{V}_i^{k+} \cap A_{m(i)}|],$$

et à tort :

$$\sum_{j \in V_i^k, H_{0j} \text{Faux}} P(H_{0j} \text{ accepté}) = E [|\widehat{V}_i^{k+} \cap A_{m(i)}^c|].$$

($A_{m(i)}^c$ est le complémentaire de $A_{m(i)}$ dans $\{1 \dots n\}^2$). L'espérance du nombre de positifs est égale au nombre moyen de pixels $j \in V_i^k$ sélectionnés :

$$\sum_{j \in V_i^k} P(H_{0j} \text{ accepté}) = E [|\widehat{V}_i^{k+}|].$$

Polzehl et Sopkoyny dans [16] cherchent à contrôler la probabilité que le pixel j soit déclaré positif à tort. Aussi,

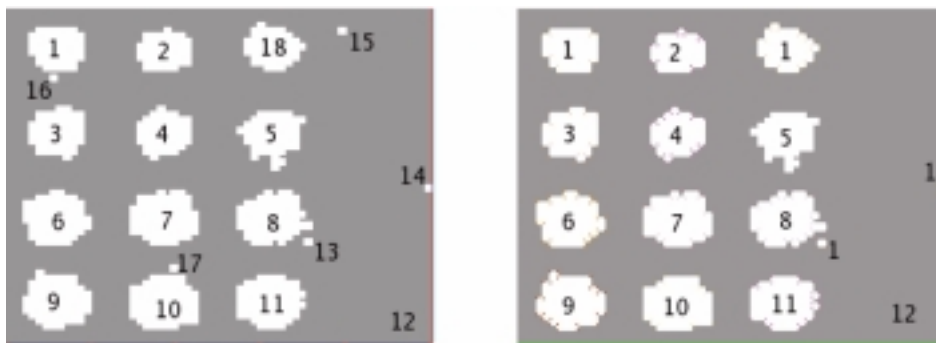


Figure 6. Carte des zones identifiées avant (à gauche : 18 zones) et après (à droite 12 zones) le regroupement de zones par minimisation d'un critère pénalisé.

leur région de rejet étant bilatérale, ils choisissent $\lambda \approx 3 \approx z_{1-0,05/2}$ ($\Phi^{-1}(\alpha_{ij}) = z_\alpha$). Dans notre cas, en supposant aussi que $T(\Delta_{ij}^k)$ est de loi normale centrée réduite sous H_{0j} , on peut choisir $\lambda_i^k = 1,69 = z_{1-0,05}$. Le choix $\lambda_i^k = 2$ donne de bons résultats. Cependant, il est encore plus intéressant, après un certain nombre d'étapes, de faire en sorte que le seuil puisse grandir afin d'espérer que la probabilité que le pixel j soit un faux positif s'approche de zéro lorsque la taille des voisinages estimés grandit. Nous avons donc choisi :

$$\lambda_i^k = \max\{2, \sqrt{2 \log(|\widehat{V}_i^{k-1+}|)}\}.$$

Cependant, l'erreur d'estimation de θ_i à l'étape k est liée à une problématique de test multiple; c'est ce qui nous a amené à chercher à contrôler la proportion de faux positifs.

A.2. Contrôle de la proportion de faux positifs

Soit i un pixel fixé, à l'étape k , $|V_i^k|$ tests sont effectués pour l'estimation de θ_i . On note FP_i^k le nombre de faux positifs, VP_i^k le nombre de vrais positifs, FN_i^k le nombre de faux négatifs, et FP_i^k le nombre de faux positifs correspondants. Ces quantités sont résumées dans le [tableau1]

Tableau 1. Quantités associées aux $|V_i^k|$ tests.

	accepté	rejeté	total
H_0 vrai	VP_i^k	VN_i^k	$ V_i^{k+} \cap A_{m(i)} $
H_0 faux	FP_i^k	FN_i^k	$ V_i^{k+} \cap A_{m(i)}^c $
total	$ \widehat{V}_i^{k+} $	$ V_i^{k+} - \widehat{V}_i^{k+} $	$ V_i^{k+} $

Dans ce cadre, l'espérance de la proportion de faux positifs parmi les positifs est notée FDR_i^k (pour False Discovery Rate) et l'espérance de la proportion de vrais négatifs parmi les négatifs est notée FNR_i^k (pour False Non-discovery Rate) :

$$FDR_i^k = E \left[\frac{FP_i^k}{FP_i^k + VP_i^k} 1_{]0; \infty[}(FP_i^k + VP_i^k) \right],$$

$$FNR_i^k = E \left[\frac{FN_i^k}{FN_i^k + VN_i^k} 1_{]0; \infty[}(FN_i^k + VN_i^k) \right].$$

Si l'image est composée de deux zones A_1 et A_2 , A_1 contenant i et sur laquelle θ_i ($i \in A_1$) vaut 0 et A_2 sur laquelle θ_j ($j \in A_2$) vaut a , et si les observations ne dépendaient pas des poids w_{ij}^k (ce qui n'est qu'une approximation), on aurait :

$$E[\widehat{\theta}_i^k - \theta_i] = a \times FNR_i^k \quad \text{et donc}$$

$$E[\Delta_{ij}^k] = a(FNR_i^k - FNR_j^k) + (\theta_i - \theta_j).$$

Le biais d'estimation est lié à l'erreur FNR_i^k . Étant donné qu'il n'est pas possible de contrôler FNR_i^k mais qu'un bon contrôle

de FDR_i^k peut amener un bon contrôle de FNR_i^k (voir [23] ou encore [2]), nous avons cherché à contrôler l'erreur FDR_i^k par une borne $q = 0.01$. Le contrôle du FDR_i^k se fait en utilisant une variante de la procédure de Benjamini, et Hochberg ([3]). En effet, nos données étant stochastiquement dépendantes, l'utilisation des résultats de Benjamini et Yekutieli ([4]) est nécessaire. D'autre part, nous avons utilisé l'interprétation Bayésienne de Storey ([20] [19] [21]) et introduit une information *a priori* sur la proportion π_i^k de vraies H_{0j} . À chaque étape k , cette information est réactualisée. Le seuil λ_i^k est finalement choisi de la façon suivante :

- Ordonner (dans l'ordre croissant) les p-valeurs $p_j = 1 - \Phi(T_{ij}^k)$, $j \in V_i^k$ ($\Phi(\cdot)$ est la fonction de répartition de la loi normale centrée réduite et $T_{ij}^k = T(\Delta_{ij}^k)$ est défini dans l'algorithme).
- Choisir

$$j_{FDR} = \max \left\{ j : p_{(j)} \leq \frac{jq}{|V_i^k| \pi_i^k \sum_{p \in V_i^k} 1/p} \right\}, \lambda_i^k = T_{ij_{FDR}}^k.$$

- À l'étape suivante estimer la proportion *a priori* d'hypothèses H_{0j} vraies par :

$$\pi_i^k = |\widehat{V}_i^{k-1+}| / |V_i^k|.$$

Si, sous H_0 , $E[\Delta_{ij}^k] = 0$ alors ce type de démarche permet en effet de contrôler le FDR_i^k par q . Il est aussi raisonnable de chercher à diminuer q avec le nombre d'étapes pour espérer obtenir un biais asymptotiquement nul. Les résultats pratiques obtenus ne sont pas bien meilleurs que ceux obtenus avec la méthode décrite au paragraphe précédent. Son coût algorithmique étant élevé, nous avons choisi, dans les applications, de n'utiliser que la première démarche.

B. Calcul et estimation de $E_0[M(\Gamma)]$

Sans perte de généralité, admettons que $u_g \in A_l$ et $u_d \in A_r$. Posons $\tilde{A}_m(u) = A_m \cap \text{Vois}(u) \setminus \{u_g, u_d\}$, l'ensemble des votants de la zone m privé de $\{u_g, u_d\}$ et $\mu_{mi} = a_m - a_i$ la différence entre la valeur de f dans la zone m et dans la zone i . Nous avons

$$E[M(u, \text{Vois}(u))] = 2P(\hat{w}_{ij} = 0) \sum_{m=1}^M \sum_{s \in \tilde{A}_m(u)} P(D_m(s)),$$

avec

$$D_m(s) = \{ "s \in A_m \text{ vote pour } u" \} = \{\hat{w}_{ik} = \hat{w}_{jk}\},$$

et puisque les votes sont finalement obtenus par le test d'hypothèse fonctionnel décrit dans la partie 3, on peut écrire :

$$D_m = \{T(\xi_m - \xi_g - \mu_{ml}) < \lambda \text{ et } T(\xi_m - \xi_d - \mu_{mr}) > \lambda\} \cup \{T(\xi_m - \xi_g - \mu_{ml}) > \lambda \text{ et } T(\xi_m - \xi_d - \mu_{mr}) < \lambda\},$$

où les ξ_i pour $i \in \{1 \dots p\}^2$ ont la même loi que ξ , vecteur aléatoire gaussien de dimension N , centré, de matrice de covariance identité. Les notations

$$\beta_\lambda(\mu) = P(T(\mu + \xi_m) > \lambda) \text{ et } \phi_\lambda(\mu_1, \mu_2) = E[\beta_\lambda(\xi_d - \mu_1)(1 - \beta_\lambda(\xi_g - \mu_2))].$$

nous permettent d'obtenir, par conditionnement par rapport à ξ_g et ξ_d , l'expression de l'espérance :

$$E[M(u, Vois(u))] = 2P(T(\mu_{gd} + \xi_g - \xi_d) > \lambda) + \sum_{m=1}^M |\tilde{A}_m(u)| (\phi_\lambda(\mu_{mr}, \mu_{ml}) + \phi_\lambda(\mu_{ml}, \mu_{mr})). \quad (17)$$

Si il n'y a pas de frontière entre u_g et u_d cela signifie que $l = r$ et par conséquent :

$$E_0[M(u, Vois(u))] = 2P(\mathcal{N}(0,1) > \sqrt{2}\lambda) + 2 \sum_{m=1}^M |\tilde{A}_m(u)| \phi_\lambda(\mu_{ml}, \mu_{ml}).$$

Cette espérance est estimée par une méthode de type plug-in :

$$\widehat{E}_0[M(u, Vois(u))] = \sum_{i \in Vois(u) \setminus \{u_g, u_d\}} (\phi_{\lambda_i^k}(\hat{\theta}_i^k - \hat{\theta}_{u_g}^k, \hat{\theta}_i^k - \hat{\theta}_{u_g}^k) + \phi_{\lambda_i^k}(\hat{\theta}_i^k - \hat{\theta}_{u_d}^k, \hat{\theta}_i^k - \hat{\theta}_{u_d}^k)) + 2P(\mathcal{N}(0,1) > \sqrt{2}\lambda).$$

(les $\hat{\theta}_i^k$ étant les estimateurs obtenus par l'algorithme AWS.)

Références

[1] F. ABRAMOVICH, A. ANTONIADIS, T. SAPATINAS, and B. VIDAKOVIC, Optimal testing in a fixed-effects functional analysis of variance model, *International Journal of Wavelets, Multiresolution and Information Processing*, 2004.
[2] F. ABRAMOVICH, Y. BENJAMINI, D. DONOHO, and I. JOHNSTONE, Adapting to unknown sparsity by controlling the false discovery rate, *Annals of statistics*, 34, 2006.

[3] Y. BENJAMINI and Y. HOCHEBERG, Controlling the false discovery rate : a practical and powerful, approach to multiple testing, *Journal of Royal Statistical Society B*, 57:289-300, 1995.
[4] Y. BENJAMINI and D. YEKUTIELI, The control of the false discovery rate in multiple testing under dependency, *The Annals of Statistics*, 29(4):1165-1188, 2001.
[5] L. BREIMAN, J. FRIEDMAN, OLSHEN R., and STONE C.J, *Classification and regression trees*, Belmont, CA: Wadsworth, 1983.
[6] D. CANET, J.C. BOUDEL, and E. CANET SOULAS, *La RMN, concepts, méthodes et applications*, Dunod, 2002.
[7] D. DONOHO, Cart and best-ortho-basis: A connection, *Annals of Statistics*, 25 : 1870--1911, 1997.
[8] J. FAN, Test of significance based on wavelet thresholding and neyman's truncation, *JASA*, 91:674-688, 1996.
[9] L. GUIGUES, *Modèles Multi-Échelles pour la Segmentation d'Images*, PhD thesis, IGN, 2003.
[10] G. HAGBERG, From magnetic resonance spectroscopy to classification of tumors, a review of pattern recognition methods, *NMR in Biomedicine*, 156 : 11 : 148, 1998.
[11] I.YU., INGSTER and I. SUSLINA, *Nonparametric Goodness-of-Fit Testing under Gaussian Model.*, volume 169 of *Lecture Notes in Statistics*, Springer-Verlag, New-York, 2002.
[12] IAN H. JERMYN and H. ISHIKAWA, Globally optimal regions and boundaries as minimum ratio weight cycles, *IEEE Transaction on pattern analysis and machine intelligence*, 23(10), oct 2001.
[13] M. KOHLER, Nonparametric estimation of piecewise smooth regression functions, Technical report, Stuttgart, 2003.
[14] S. MALLAT, *A Wavelet Tour of Signal Processing*, Academic Press, 1999.
[15] H. PHILIPPE, N. STRANSKY, J.P. THIERY, F. RADVANYI, and E. BARILLOT, Analysis of array cgh data: from signal ratio to gain and loss of dna regions, *Bioinformatics*, 20(18) : 3413-3422, 2004.
[16] J. POLZEHL and V. SPOKOINY, Adaptive weights smoothing with applications to image restoration, *J.R Stat Soc B*, 62 : 335-354, 2000.
[17] J. POLZEHL and V. SPOKOINY, Vector adaptive weights smoothing with application to mri, *J.R Stat Soc B*, 63 : 335-354, 2001.
[18] V. SPOKOINY, Adaptive hypothesis testing using wavelets, *Annals of Statistics*, 24(6):2477--2498, december 1996.
[19] J.D. STOREY, The positive false discovery rate: A bayesian interpretation and the q-value, *Annals of Statistics*, 31 : 2013-2035, 2003.
[20] J.D. STOREY, J.E. TAYLOR, and SIEGMUND, A direct approach to false discovery rates, *Journal of the Royal Statistical Society, Series B*, 64 : 479-498, 2002.
[21] J.D. STOREY, J.E. TAYLOR, and SIEGMUND, Strong control, conservative point estimation, and simultaneous conservative consistency of false discovery rates: A unified approach, *Journal of the Royal Statistical Society, Series B*, 66 : 187-205, 2004.
[22] FABIEN SZABLO DE EDELENYI, *Développement d'une nouvelle approche d'analyse des images spectroscopiques RMN: les images nosologiques*, PhD thesis, UJF, novembre 2001.
[23] J. TAYLOR, R. TIBSHIRANI, and B. EFRON, The miss rate for the analysis of expression data, *Biostatistics*, 6(1) : 111-117, 2005.
[24] B. WHITCHER, A.J. SCHWARZ, H. BARJAT, S. SMART, R. GRUNDY, and M. F. JAMES, Wavelet-based cluster analysis: Data-driven grouping of voxel time-courses with application to perfusion-weighted and pharmacological mri of the rat brain, *NeuroImage*, 24(2) : 281-295, 2005.
[25] C. ZHU, S. and A. YUILLE, Region competition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(9), 1996.



Robin Girard

Robin Girard est étudiant en thèse au Laboratoire LMC à Grenoble. Sa thèse s'effectue au sein d'un projet financé par la région Rhones alpes dans le cadre de la thématique cancer. Il s'intéresse à l'analyse de données de grande dimension et plus particulièrement à l'étude mathématique d'algorithmes basés sur des principes statistiques et visant à analyser des images hyper-spectrales.

