Projet: Principal Curve Clustering with Noise

DATA MINING

MASTER 2 ISIFAR – Mars 2020

Insa BADJI

Khayreddine NACEUR

Christian RAMIRES ALIAGA

I. INTRODUCTION

Le but de ce projet est de détecter automatiquement des entités (clusters ou groupes) curvilignes dans les processus ponctuels spatiaux. Les processus ponctuels spatiaux sont divisés en deux catégories de points : les points de bruit et les points caractéristiques. Les points de bruit sont des points observés qui ne font pas partie des clusters et les points caractéristiques sont ceux qui sont regroupés sous forme de trajectoires curvilignes. Les domaines d'application de cette technique sont entre autres la détection dans les images de reconnaissance aérienne des champs de mines curvilignes et la détection de banquises. La figure 1. (a) est une simulation d'une telle image et la figure 1. (b) montre les clusters curvilignes détectés par notre méthode.

Dans la section II, nous donnons quelques informations sur les courbes principales et introduisons notre modèle de probabilité et notre algorithme de clustering. La section II.3 présente notre méthode de regroupement sur les courbes principales ouvertes et la section II.4 décrit l'utilisation des facteurs de Bayes approximatifs pour choisir le nombre de caractéristiques et la quantité de lissage simultanément et de façon automatique. Les méthodes d'initialisation, y compris notre algorithme de clustering de courbe principale hiérarchique (HPCC), sont discutées dans la section III. Des exemples sont présentés dans la section IV et, dans la section V, nous discutons sur tout ce que nous avons traité dans ce projet.

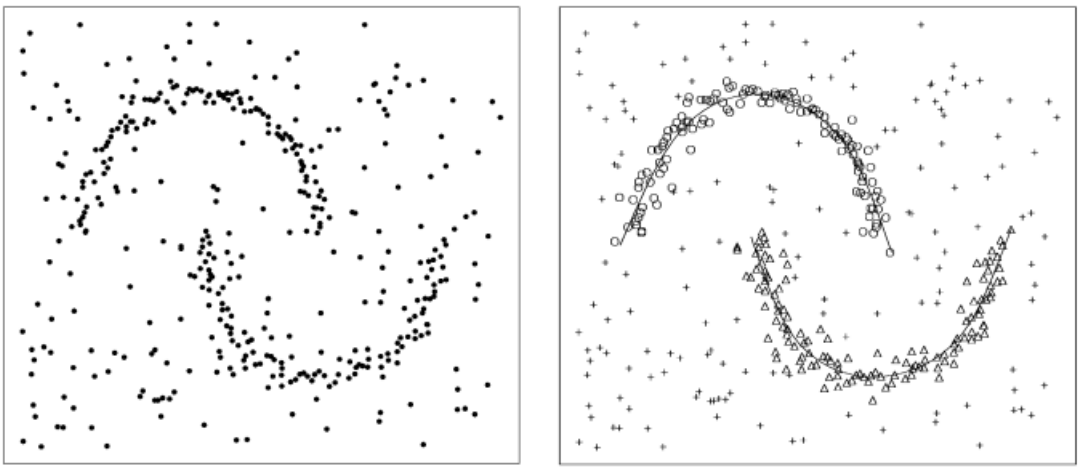

Fig. 1. (a) données simulées, (b) résultat final

II. MODÈLE, ESTIMATION ET INFÉRENCE

II.1 Courbes principales

Définition : (Courbe auto-consistante). La courbe paramétrée $f : \mathrm{I} \longrightarrow \mathbb{R}^d$ est dite auto consistante pour X si, pour presque tout λ,

$$f(\lambda) = \mathbb{E}[\mathrm{X}|\lambda_f(\mathrm{X}) = \lambda] \qquad (1)$$

Une courbe principale est définie de la manière suivante :

Une courbe paramétrée f de classe C^∞ est une courbe principale pour X si elle est sans point double (c'est-à-dire une courbe injective), de longueur finie à l'intérieur de toute boule de $\mathbb{R}^d$ et auto-consistante. Pour un ensemble de données, l'auto-consistance peut s'interpréter en disant que chaque point de la courbe f est la moyenne des observations qui se projettent sur ce point. Les courbes principales ont été introduites par *Hastie et Stuetzle* et discutées dans le contexte du clustering par *Banfield et Raftery*.

L'algorithme d'ajustement d'une courbe principale à partir de données implique l'application itérative de la définition (1), où l'espérance conditionnelle est remplacée par un nuage de points plus lisse. Le choix du paramètre de lissage est examiné à la section 2.4. Chaque point de données x_j , a un point de projection associé $f(\lambda_j)$ sur la courbe, qui est le point de la courbe le plus proche de x_j (voir figure 2). Le segment de droite de x_j à $f(\lambda_j)$ est orthogonal à la courbe en $f(\lambda_j)$ sauf si $f(\lambda_j)$ est un point final de la courbe. La correction du biais pour les courbes principales fermées peut également être étendue aux courbes principales ouvertes que nous utilisons ici.

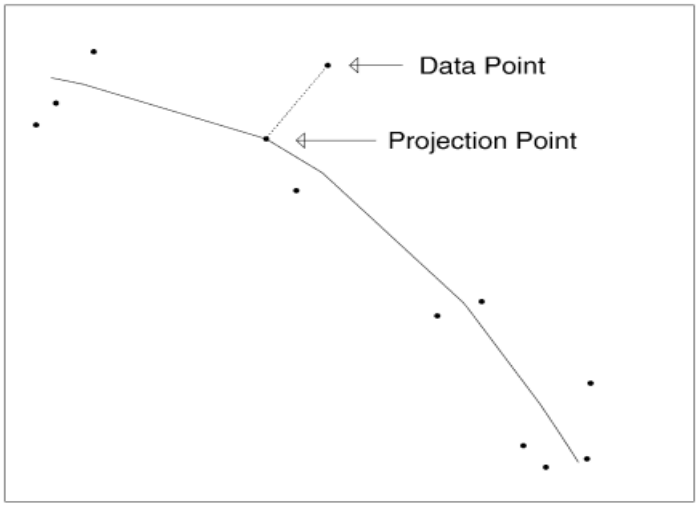

Fig. 2. Courbe principale et ses projections

II.2 Modèle de probabilité

Il s'agit de modéliser un processus ponctuel spatial avec bruit de fond. Il faut noter que les données sont réparties en deux familles (les points de bruit et les points caractéristiques) et que des hypothèses de distribution sont faites sur les données. Supposons que les données X sont un ensemble d'observations $x_1, \ldots, x_N$ et C une partition des données en K+1 clusters $(C_0, C_1, \ldots, C_K)$. Chaque cluster C_j contient N_j observations, C_0 représente le groupe des points de bruit et les autres groupes représentent les points caractéristiques. Les points du groupe C_0 sont supposés répartis uniformément sur la région de l'image alors que les points caractéristiques quant à eux sont supposés être distribués le long de leur vrai groupe caractéristique, ce qui veut dire que leurs points de projection sur la courbe principale de leur groupe caractéristique suivent la loi uniforme $U(0, \nu_j)$ où ν_j est la longueur de la courbe principale du groupe C_j. Nous supposons aussi que les points caractéristiques sont distribués sur leur groupe sous-jacent selon la loi normale centrée et de variance ${\sigma_j}^2$. La distance d'un point à une courbe est la distance de projection orthogonale la plus courte du point à la courbe. Si le point se projette sur un point extrême de la courbe, la distance est dans ce cas la longueur entre le point et ce point extrême. Le modèle ci-présent est considéré comme un modèle de mélange de K+1 composantes et π_j (j $= 0, 1 \ldots K$) les proportions associées, c'est-à-dire les probabilités d'appartenance à la composante C_j. Le modèle conduit donc à l'estimation du paramètre $\theta = (\pi_0, \pi_1, \ldots, \pi_K, \nu_1, \ldots, \nu_K, {\sigma_1}^2, \ldots, {\sigma_K}^2)$. La vraisemblance est donnée ci-dessous :

$$L(X|\theta) = \prod_{i=1}^{N} L(x_i|\theta)$$

Tel que :

$$L(x_i|\theta) = \sum_{j=0}^{K} \pi_j L(x_i|\theta, x_i \in C_j)$$

Et pour j $= 1 \ldots K$, nous avons :

$$L(x_i|\theta, x_i \in C_j) = \left[\frac{1}{\nu_j}\right]\left[\frac{1}{\sqrt{2\pi}\sigma_j}\exp\left(\frac{-\|x_i - f(\lambda_{ij})\|^2}{2\sigma_j^2}\right)\right]$$

et

$$L(x_i|\theta, x_i \in C_0) = \frac{1}{\text{Aire}}$$, où Aire est l'aire de l'image.

II. 3 Estimation : l'algorithme CEM-PCC

L'algorithme CEM-PCC a pour but d'améliorer un clustering donné en utilisant l'algorithme de Classification EM [6], qui est une version de l'algorithme EM bien connu, et le modèle de probabilité de la section II.2. Nous commençons par un regroupement initial des méthodes discutées dans la section III.

Présentation de l'algorithme CEM-PCC :

1. Commencer par un clustering initial (les points caractéristiques et les points de bruit de fond).
2. **(Étape M)** Sous réserve du clustering actuel, ajuster une courbe principale de chaque groupe de caractéristiques, puis calculer des estimations des paramètres ν_j, ${\sigma_j}^2$ et π_j.
3. (**Étape E)** En fonction des courbes actuelles et des estimations des paramètres, calculer la probabilité que chaque point se trouve dans chaque grappe.
4. **(Étape de classification)** Reclasser chaque observation dans son groupe le plus probable.
5. Vérifier la convergence ; fin de l'algorithme ou retour à l'étape 2.

Une fois que nous avons calculé la probabilité que chaque point se trouve dans chaque cluster sur la base des estimations des paramètres à l'itération actuelle, nous reclassons chaque point dans le cluster pour lequel il a la plus forte vraisemblance. À la fin de chaque itération, nous calculons la vraisemblance globale $L(\mathrm{X}|\theta)$. Ce processus est exécuté pour un nombre prédéterminé d'itérations, moment auquel nous choisissons, comme résultat final, le regroupement avec la probabilité globale la plus élevée (les itérations CEM diminuent parfois la vraisemblance).

Nous avons constaté qu'il est important d'imposer une borne inférieure à l'estimation de la variance autour de la courbe. Si la variance peut diminuer sans limite, la vraisemblance peut augmenter sans limite. Cela peut être un problème lorsqu'il y a de petits clusters car le lissage est presque capable d'interpoler les points de données. Nous imposons une borne basée sur l'hypothèse que les données ne sont pas connues avec une précision absolue. Par exemple, si nous supposons que les données sont précises à trois chiffres significatifs, alors nous pouvons trouver une borne inférieure sur la résolution des données et la traduire en borne inférieure sur la variance.

II. 4 Inférence : Choix simultané du nombre de clusters et de leur lissage

Le choix du nombre de clusters et le paramètre global de lissage se font simultanément car le nombre de clusters a un effet sur le paramètre de lissage. Nous allons dans ce cas utiliser des *B-splines* cubiques comme courbes principales, ce qui dans la pratique se fait par l'utilisation de la fonction *principal_curve* qui à son tour fait appel à la fonction *smooth_spline*. Le niveau de lissage dans chaque cluster est mesuré par le degré de liberté de sa courbe principale ajustée.

Chaque combinaison d'un nombre de clusters et d'un degré de liberté peut être considéré comme un éventuel modèle de données et les modèles peuvent être comparés par le facteur de Bayes. Le *facteur de Bayes* qui sépare deux hypothèses ou deux « modèles » M1 et M2, est une mesure de leur mérite relatif, le rapport de leurs vraisemblances : $\frac{\mathbb{P}(\mathrm{X}|\mathrm{M1})}{\mathbb{P}(\mathrm{X}|\mathrm{M2})}$, c'est-à-dire le rapport de leurs probabilités à postériori lorsque leurs probabilités a priori sont égales. Les facteurs de bayes sont attrayants

dans le contexte actuel parce que, contrairement aux tests de significativité statistique standard, ils nous permettent de bien comparer les modèles non imbriqués que nous considérons et ils permettent également une bonne comparaison plusieurs modèles.

Nous faisons une approximation du facteur de Bayes en utilisant le critère d'information bayésien (*Bayesian Information Criterion BIC*). Dans les modèles réguliers, la différence entre les valeurs du BIC pour deux modèles est approximativement égale au double du log du facteur de Bayes lorsque les informations d'unité a priori pour les paramètres du modèle sont utilisées. La sélection de modèle basée sur le critère du BIC fournit des estimateurs asymptotiquement constants des paramètres de la loi de distribution des données. Cette approche s'est avérée bien fonctionner dans la pratique pour les modèles de mélange.

Le BIC pour un modèle de K groupes de points caractéristiques et d'un groupe de points de bruit est définie par :

$$BIC = 2\log\big(L(X|\theta)\big) - M \cdot \log(N)\,,$$

avec $M = K(DF + 2) + K + 1$ le nombre total de paramètres à estimer. Ce qui est assez simple à remarquer car pour chacun des K groupes de points caractéristiques nous estimons ν_j, ${\sigma_j}^2$ et le degré de liberté DF de courbe principale ajustée. On y ajoute aussi les proportions de ces K groupes et une valeur estimée de l'aire de la région d'image.

III. INITIALISATION

III. 1 Débruitage et regroupement initial

La performance de l'algorithme CEM-PCC peut être sensible à la valeur de départ, il est donc important d'avoir une bonne valeur de départ. Le clustering initial utilisé pour obtenir cette valeur devrait permettre d'atteindre deux objectifs : séparer les points caractéristiques des bruits de fond et fournir un clustering initial des points d'observation. La première peut être réalisée à la main ou par diverses méthodes automatiques, telles que le maximum de vraisemblance non paramétrique, en utilisant la *tessellation de Voronoï* ou l'élimination du K^{ème} plus proche voisin. Cette étape n'a pas besoin d'être exhaustive puisque le CEM-PCC examinera les points de bruit pour déterminer s'ils doivent être inclus dans les clusters et vice-versa.

Une fois que les points de bruit supprimés, nous procédons à un premier regroupement des points caractéristiques afin de pouvoir ajuster une courbe principale à chaque groupe. Nous exigeons qu'il y ait au moins sept points dans chaque cluster ; lorsqu'il y en a moins de sept, nous ajustons une ligne droite de composante principale au lieu d'une courbe. Les points caractéristiques peuvent être regroupés en utilisant un modèle de clustering basé sur le modèle, tel qu'il est mis en œuvre dans la fonction *MCLUST* de R.

III. 2 Classification hiérarchique en courbe principale (HPCC)

La classification sur les courbes principales fermées a été introduite par *Banfield and Raftery*. Le critère de regroupement (V*) est basé sur une somme pondérée des distances au carré autour de la courbe et des distances au carré le long de la courbe et ils stipulent que ce critère est optimal lorsque les points sont normalement répartis autour de la courbe (conditionnellement aux courbes estimées et en choisissant un bon α). La valeur V^* est définie comme suit :

$$V^* = V_{About} + \alpha V_{Along} ,$$

où

$$V_{About} = \sum_{j=1}^{N} \|x_j - f(\lambda_j)\|^2$$

$$V_{Along} = \frac{1}{2} \sum_{j=1}^{N} \|\epsilon_j - \bar{\epsilon}\|^2$$

et

$$\epsilon_j = f(\lambda_j) + f(\lambda_{j+1})$$

Le terme V_{About} mesure la dispersion des observations autour de la courbe (en distance orthogonale à la courbe), tandis que le terme V_{Along} lui mesure la variance des distances de longueur d'arc entre les points de projection sur la courbe. Minimiser la somme ΣV^* sur tous les clusters conduira à des clusters avec des points régulièrement espacés le long de la courbe et étroitement groupés autour d'elle. Des valeurs élevées d'alpha feront en sorte que l'algorithme évite les clusters avec des écarts le long de la courbe principale, tandis que les petites valeurs favoriseront les clusters plus fins. Le regroupement s'arrête lorsque la fusion de clusters entraîne une augmentation de la somme ΣV^*.

La méthode sur les courbes principales fermées sera étendue sur les courbes principales ouvertes en changeant V_{Along} de sorte que l'on somme uniquement jusqu'à (N-1) plutôt que N. Ceci est dû au fait qu'une courbe fermée est une courbe qui se rejoint elle-même, tandis que la courbe ouverte s'arrête à ses points d'extrémité.

Donnons une présentation de l'algorithme HPCC :

1. Faire une première estimation des points de bruit et les supprimer.
2. Former un regroupement initial avec au moins sept points dans chaque cluster.
3. Ajuster une courbe principale à chaque cluster.
4. Calculer ΣV^* pour chaque fusion possible.
5. Effectuer la fusion qui conduit à ΣV^* la plus faible.
6. Continuer à fusionner jusqu'à ce que le nombre souhaité de clusters soit atteint.

Cette approche est plus appropriée pour les courbes fermées dans la mesure où il est plus difficile de décider quand arrêter le regroupement pour les courbes ouvertes. Dans le cas des courbes fermées, le regroupement s'arrête lorsqu'une fusion entraîne une augmentation de la somme de V^*. Nous avons surmonté ce problème en utilisant des facteurs Bayes approximatifs.

IV. EXEMPLE

Dans cette partie, nous allons appliquer la méthode du « ***PCC with noise*** » sur une base de données que nous avons choisi sur le site CERI (*Centre for Earthquake Research and Information*) qui met à disposition des nombreuses informations sur les tremblements de terre. Plus particulièrement, notre base de données contient 1034 observations de tremblements de terre sur *la plaque de la mer des Philippines* dont les caractéristiques sont la latitude et la longitude. Nous avons sélectionné ceux-ci parmi les observations entre 2010 et 2020 car sa distribution nous a semblé pertinente pour effectuer notre étude.

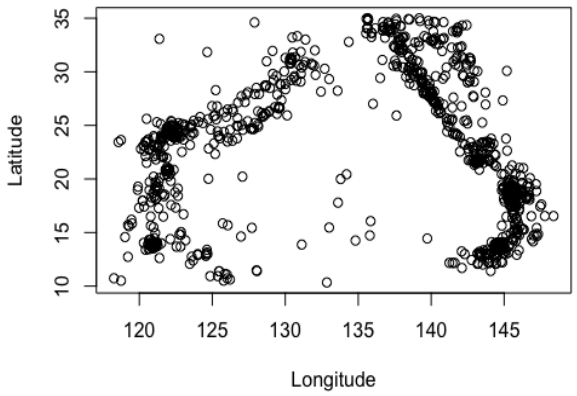

Fig. 3 Distribution de toutes observations.

La méthode du « ***PCC with noise*** » se divise en 2 parties :

1. La méthode HPCC qui après débruitage de l'information fusionne des clusters minimisant la somme $\sum V*$ à partir d'un clustering initial sur lequel des courbes principales ont été ajustées.

2. L'algorithme CEM-PCC qui à chaque itération, reclasse chaque observation dans le cluster le plus vraisemblable après estimation des paramètres.

Dans la suite, nous allons détailler chacune de ces parties.

IV. 1 La méthode HPPC

a) Débruitage

Comme nous l'avons exposé précédemment la première étape de l'algorithme consiste à diviser l'ensemble des individus en deux groupes : les points caractéristiques (*features points*) et les points de bruit de fond (*background noise*). En appliquant la fonction *NNclean* du package « prabclus », nous avons fait le débruitage de l'ensemble de nos observations et nous obtenons la figure ci-dessous.

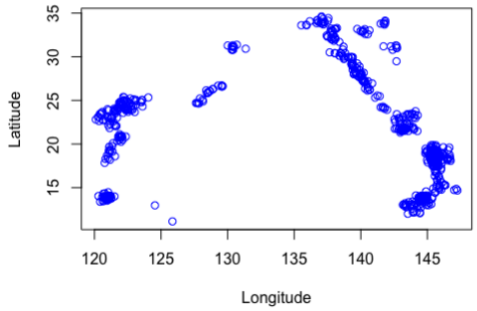

Fig. 4 La base de données sans bruit de fond.

b) Clustering des points caractéristiques

Nous avons constaté que autour de certains points un grand nombre d'observations étaient concentrées. Alors, nous avons utilisé la fonction *Mclust* du package « *mclust* » basée sur un modèle de mélange gaussien paramétrique pour obtenir 9 clusters :

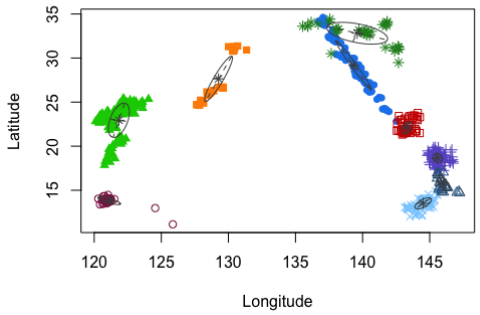

Fig. 5 Clustering initial en utilisant Mclust. Il y a 9 clusters.

c) Fusion des clusters

Pour la suite de l'algorithme, nous avons utilisé et modifié le code de *Derek Stanford et Adrian Raftery* car certaines fonctions étaient obsolètes, comme par exemple, la fonction « *principal_curve* » pour tracer la courbe principale.

Comme nous l'avons expliqué précédemment, la méthode du « *PCC with noise* » dépend du critère BIC qui est déterminé en connaissant le nombre de clusters et le nombre de degrés de liberté. C'est pour cette raison que nous avons fait un tableau (voir p. 11) contenant ces informations et nous avons constaté que la plus grande valeur du BIC correspondait à 3 clusters avec 8 degrés de liberté. Ainsi, nous avons appliqué l'algorithme *HPPC* des auteurs de l'article pour K = 3 et DF = 8 afin d'obtenir ce graphique.

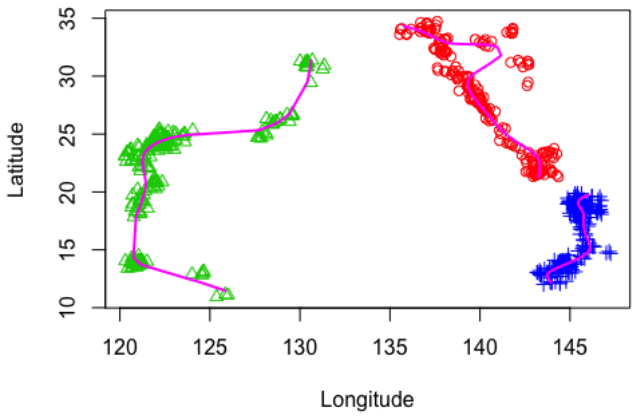

Fig. 6 Trois clusters avec ses courbes principales.

IV. 2 L'algorithme CEM-PCC

Nous utilisons le clustering obtenu à l'étape précédente auxquels nous rajoutons les bruits de fond – souvent nommé cluster C_0 – pour former le clustering initial de l'algorithme CEM-PCC. Ceci fait une reclassification de toutes les observations en les distribuant dans les 3 clusters caractéristiques et le cluster C_0, ainsi que l'ajustement des courbes principales. Nous pouvons apprécier le résultat final ci-dessous :

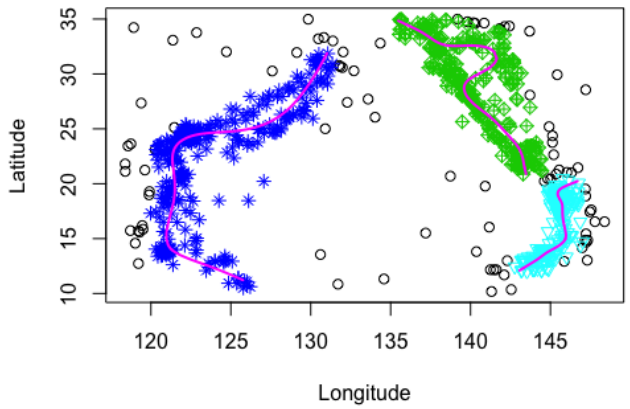

Fig. 7 Résultat final de l'algorithme CEM-PCC.

Tableau représentant les valeurs du BIC en fonction du nombre de degrés de liberté et du nombre des clusters caractéristiques.

DF	0 clusters	1 cluster	2 clusters	3 clusters	4 clusters	5 clusters	6 clusters	7 clusters	8 clusters	9 clusters
2	-13682.2	-12571.71	-11618.61	-11370.71	-11413.77	-11198.15	-11135.38	-11079.94	-10867.42	-10812.69
3	-13682.2	-12542.33	-11659.21	-11501.35	-10979.95	-10952.31	-10801.35	-10772.39	-10714.03	-10704.85
4	-13682.2	-12101.19	-11250.09	-11044.24	-11027.37	-10753.10	-10623.73	-10552.93	-10585.86	-10576.66
5	-13682.2	-11768.66	-11087.71	-10762.47	-10793.18	-10685.11	-10637.81	-10554.27	-10496.18	-10523.61
6	-13682.2	-11553.72	-10902.25	-10576.62	-10683.38	-10671.42	-10617.00	-10541.62	-10472.85	-10510.01
7	-13682.2	-11440.80	-10792.26	-10530.15	-10678.11	-10676.45	-10561.76	-10515.57	-10582.76	-10528.28
8	-13682.2	-11377.20	-10711.61	**-10383.11**	-10552.25	-10573.10	-10561.16	-10520.74	-10480.46	-10548.73
9	-13682.2	-11314.15	-10631.64	-10817.03	-10830.27	-10698.78	-10599.83	-10576.66	-10647.68	-10557.54
10	-13682.2	-11312.85	-10547.84	-10750.99	-10753.89	-10603.56	-10583.37	-10590.90	-10683.01	-10597.69
11	-13682.2	-11256.00	-10481.91	-10697.01	-10697.13	-10571.47	-10514.33	-10559.92	-10665.44	-10637.80
12	-13682.2	-11253.57	-10436.42	-10655.67	-10661.77	-10546.87	-10504.27	-10541.04	-10672.01	-10673.40
13	-13682.2	-11302.86	-10436.01	-10662.90	-10685.58	-10545.95	-10608.52	-10567.75	-10604.61	-10716.64
14	-13682.2	-11289.99	-10547.85	-10769.96	-10489.50	-10533.35	-10524.04	-10566.22	-10638.94	-10770.08
15	-13682.2	-11348.06	-10537.01	-10766.80	-10849.32	-10552.59	-10551.36	-10618.62	-10648.33	-10789.12
16	-13682.2	-11300.41	-10498.53	-10736.40	-10829.56	-10558.33	-10576.29	-10649.43	-10681.12	-10804.00
17	-13682.2	-11392.65	-10450.69	-10697.66	-10796.89	-10558.73	-10540.09	-10675.07	-10716.82	-10847.54
18	-13682.2	-11368.62	-10405.10	-10662.11	-10766.57	-10540.46	-10540.49	-10683.72	-10762.54	-10863.81
19	-13682.2	-11371.58	-10706.78	-10634.79	-10744.18	-10520.93	-10591.03	-10685.14	-10787.43	-10956.93
20	-13682.2	-11454.21	-10434.65	-10632.29	-10740.81	-10505.49	-10546.59	-10711.36	-10823.14	-10982.20

V. CONCLUSION

Nous avons détecté automatiquement des clusters curvilignes des données avec bruit de fond issues d'images de reconnaissance aérienne par la combinaison de la modélisation paramétrique du bruit représentée par l'algorithme CEM-PCC et la modélisation non paramétrique de la forme des groupes par l'algorithme HPCC. Nous avons ainsi mis en application ces deux algorithmes sur des données de tremblements de terre des *plaques* tectoniques *de la mer des Philippines* après avoir fait le choix de la meilleure combinaison du nombre de clusters et degré de liberté des données (c'est-à-dire le meilleur modèle de mélange parmi ceux en compétition). Le meilleur modèle obtenu est celui avec trois clusters curvilinéaires avec un degré de liberté égale à 8.