



HAL
open science

Méthodes multi-modèles de co-recalage de surfaces : Application à l'IRM à très haut champ (7 Tesla) de l'hippocampe humain

Claire Cury

► **To cite this version:**

Claire Cury. Méthodes multi-modèles de co-recalage de surfaces : Application à l'IRM à très haut champ (7 Tesla) de l'hippocampe humain. Imagerie. 2011. hal-01672360

HAL Id: hal-01672360

<https://inria.hal.science/hal-01672360v1>

Submitted on 24 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Rapport de stage Master 2 Imagerie médicale

Méthodes multi-modèles de co-recalage de surfaces : Application à l'IRM à très haut champ (7 Tesla) de l'hippocampe humain

Laboratoire d'accueil : CRICM UMR7225 /UMR-S 975. Equipe COGIMAGE

Claire CURY

Encadrants : Joan Glaunès et Olivier Colliot

3 septembre 2011

Remerciements

Ce stage de Master 2 a été effectué au Centre de Recherche de l'Institut du Cerveau et de la Moelle Epinière (CRICM) dans l'enceinte de l'hôpital Pitié-Salpêtrière, à Paris, au sein de l'équipe COGIMAGE (Neurosciences Cognitives et Imagerie Cérébrale). Je tiens en premier lieu à remercier mes encadrants Joan Glaunès et Olivier Colliot. Je les remercie de m'avoir fait découvrir et apprécier le monde de la recherche, ainsi que pour leurs conseils, leur patience (en particulier Joan face à mes nombreuses questions mathématiques souvent répétitives). Je les remercie aussi, pour leurs disponibilités, d'avoir pris le temps de relire mon rapport et leur soutien constant. Je tiens également à leur exprimer ma reconnaissance, pour m'avoir permise de continuer ces travaux en thèse.

Je tiens aussi à remercier l'ensemble de l'équipe pour son accueil son esprit de groupe et sa bonne humeur.

Table des matières

1	Introduction	7
2	Motivations et état de l'art	11
2.1	Notion de variabilité anatomique	11
2.2	Recalage d'images cérébrales	12
2.3	Une peu d'histoire : Talairach, MNI et ICBM	12
2.4	Méthodes de construction d'un atlas adaptatif	13
2.5	Méthodes de modèles multiples	15
2.6	Méthodes continues d'analyse de la variabilité	15
3	Modèles de grandes déformations et recalage de surfaces	17
3.1	Espaces de Hilbert à Noyaux Reproductibles : R.K.H.S.	17
3.2	Grandes déformations difféomorphiques	18
3.3	Courants	19
3.4	Estimation d'un template à partir d'une population [15]	21
4	Données d'hippocampes à très haute résolution spatiale	23
4.1	Segmentation	23
4.2	Extraction de squelette	23
4.3	Les données synthétiques	24
4.3.1	Courbes gaussiennes	24
4.3.2	Squelettes d'hippocampes synthétiques	24
5	Approche itérative de regroupement et de recalage simultané	29
5.1	Intuition	29
5.2	Le problème : les distances	30
5.3	Quelques solutions	30
5.4	Matching Pursuit	33
5.5	Résultats	36
5.6	Discussion	40
6	Analyse en Composantes Principales des vecteurs moments initiaux	41
6.1	Les vecteurs moments initiaux	41
6.2	L'Analyse en Composantes Principales à noyau	42
6.3	Résultats	44
6.4	Discussion	45
7	Méthodes utilisant une matrice de distances	51
7.1	Estimation de la matrice de distance	51
7.2	Spectral Clustering	53
7.2.1	Méthode	53
7.2.2	Résultats	54
7.2.3	Discussion	57
7.3	ISOMAP	58
7.3.1	Méthode	58
7.3.2	Résultats	58
7.3.3	Discussion	59
8	Conclusion	61

1 Introduction

L'hippocampe est une structure du lobe temporal du cerveau qui joue un rôle considérable dans les processus cognitifs fondamentaux comme la mémoire et la navigation spatiale et qui est impliqué dans différentes maladies neurologiques comme la maladie d'Alzheimer ou les épilepsies du lobe temporal. L'analyse morphologique de l'hippocampe permet de mieux comprendre les processus pathologiques associés à ces maladies et de mieux les diagnostiquer.

L'hippocampe se situe dans le lobe temporal médian de chaque hémisphère. C'est une structure allongée d'environ 3 cm^3 , faisant 4 à 4,5 cm de long et 1 à 2 cm de large. La partie antérieure, qui est plus large, est appelée la tête, tandis que le corps de l'hippocampe est plus fin et s'allonge vers l'arrière du cerveau dans une orientation sagittale, enfin la queue est la partie la plus en arrière et remonte avec une orientation axiale. L'anatomie de l'hippocampe est complexe et constituée de différentes sous-structures internes dont les principales sont le gyrus dentelé, le subiculum et la corne d'Ammon (Figure 1).

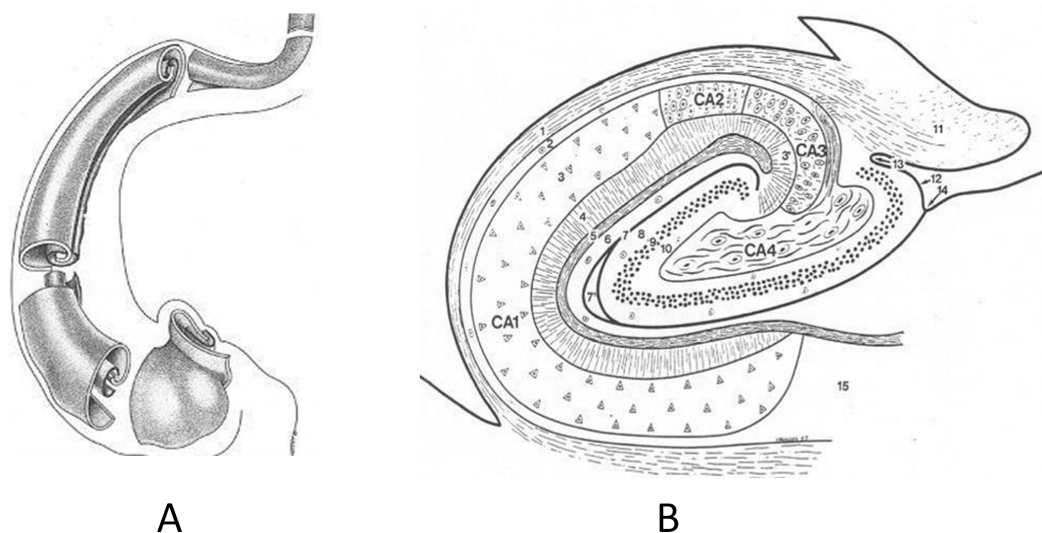


FIG. 1: Anatomie de la structure interne de l'hippocampe (d'après Duvernoy, 1998). (A) La structure interne de l'hippocampe est constituée de deux lames de substance grises enroulées l'une sur l'autre : la corne d'Ammon et le gyrus dentelé. (B) Schéma en coupe coronale de la structure interne de l'hippocampe. Les différents champs de la corne d'Ammon sont CA1-CA4. On peut également observer l'alveus (1), la fimbria (11), le subiculum (15), et le gyrus dentelé (8-9-10).

Sur les images à résonance magnétique (IRM) conventionnelles, la structure interne de l'hippocampe n'est pas visible. L'utilisation d'IRM à très haut champ (ici à 7 Tesla, le Tesla étant l'unité de mesure du champ magnétique) offre un rapport signal sur bruit accru et de nouveaux contrastes, et permet d'obtenir de hautes résolutions spatiales (de l'ordre de la centaine de microns). Dans le cas de l'hippocampe, cela permet de visualiser la structure interne et de constater une grande variabilité anatomique, notamment au niveau de la tête qui fait apparaître un nombre très variable de bosses ou plus qu'on appelle digitations (voir Figure 3).

Le stage s'inscrit dans le cadre d'un important projet visant à modéliser les interactions entre l'hippocampe et le cortex pour l'étude du fonctionnement normal et pathologique de la mémoire. Pour cela, il est nécessaire de construire des atlas de la structure interne de l'hippocampe, du fait de sa variabilité, afin de pouvoir par la suite étudier la corrélation entre l'anatomie et la cognition.

Ce stage vise à développer des méthodes pour l'analyse de la variabilité anatomique de la structure interne de l'hippocampe. En particulier, nous cherchons à développer et tester des méthodes d'analyse

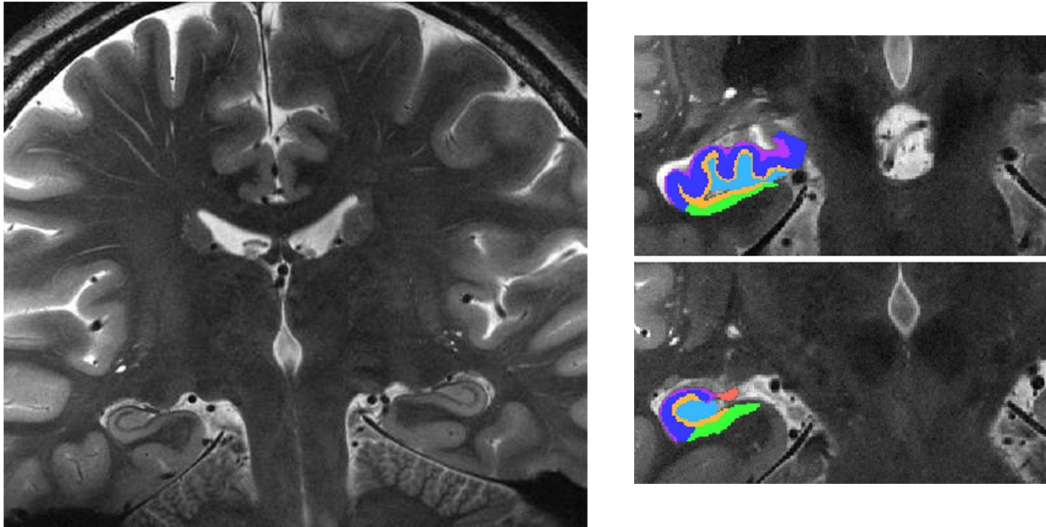


FIG. 2: *Imagerie à très haute résolution spatiale de l'hippocampe par IRM à 7T (Chupin et al., 2009 [8]). A gauche, coupe coronale d'une IRM 7T pondérée en T2 à très haute résolution spatiale ($250\mu\text{m} \times 250\mu\text{m} \times 1.3\text{mm}$). A droite, segmentation manuelle des différentes parties de la structure interne de la formation hippocampique (bleu foncé : corne d'Ammon, bleu ciel : CA4 et gyrus dentelé - hilum, vert : subiculum, orange : substance blanche de la corne d'Ammon, rouge : fimbria, violet : alveus).*

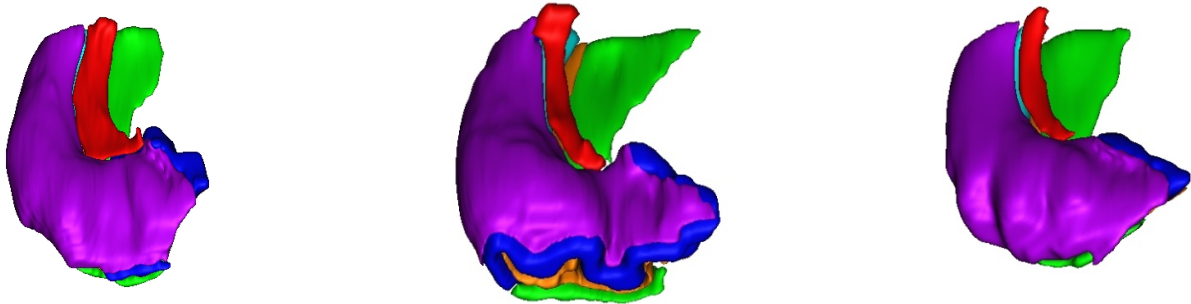


FIG. 3: *Chupin et al., 2009 [8]. Rendu 3D des segmentations manuelles pour trois sujets sains. On peut observer l'importante variabilité inter-individuelle, en particulier au niveau des digitations.*

multi-modèles, par co-recalage entre surfaces, basées sur le concept mathématique de « courant ». Ce type de méthodes est déjà utilisé pour le recalage et le co-recalage depuis plusieurs années [17]. Le but est d'étendre ce genre de méthode au problème de l'estimation multi-modèles, dans lequel une collection de surfaces observées est expliquée par un petit nombre de surfaces-modèles types. Cette extension est nécessaire lorsqu'une variabilité morphologique qualitative est observée entre les individus, comme dans le cas de l'hippocampe. Dans l'idéal, la méthode devrait donner n templates représentatifs des variations de la structure interne de la corne d'Ammon (cf figure 3) dans le cas bien sûr où cette variabilité anatomique peut être caractérisée par plusieurs modes de variations.

Au cours de ce stage, plusieurs pistes ont été explorées qui peuvent être regroupées en trois grandes familles d'approches (figure 4). La première vise à estimer directement plusieurs modèles à partir de la population (branche de droite de la figure 4). C'est une approche itérative qui nécessite plusieurs calculs de modèles qui évoluent à chaque itération. La seconde approche repose sur l'analyse de déformations de chaque individu depuis un modèle unique (branche du milieu de ce même graphique). Spécifiquement, on étudie les vecteurs moments initiaux des déformations depuis ce modèle unique, pour analyser leur répartition au sein de notre population. Enfin, la dernière approche repose sur des recalages deux-à-deux entre les individus de notre population (branche de gauche de la

figure 4). Ces recalages permettent d'obtenir une matrice de similarité entre les différents individus sur laquelle sont appliqués des algorithmes de réduction de dimension pour pouvoir classer ensuite les individus dans le nouvel espace.

Le plan de ce rapport est le suivant. La partie 2 propose un bref état de l'art ainsi que l'intérêt des approches multi-modèles. La partie 3 décrit la méthode de recalage difféomorphique de surfaces conduisant à un template qui sera par la suite très utilisée, ainsi que les outils mathématiques nécessaires à sa compréhension. Nous présenterons ensuite, dans la partie 4, les données synthétiques et réelles qui serviront à nos expériences. Enfin, nous exposerons les différentes approches étudiées. L'approche de type itératif sera présentée à la partie 5. La seconde, basée sur l'estimation d'un modèle unique, sera développée dans la partie 6. Enfin celle qui utilise des co-recalages deux à deux est décrite à la partie 7.

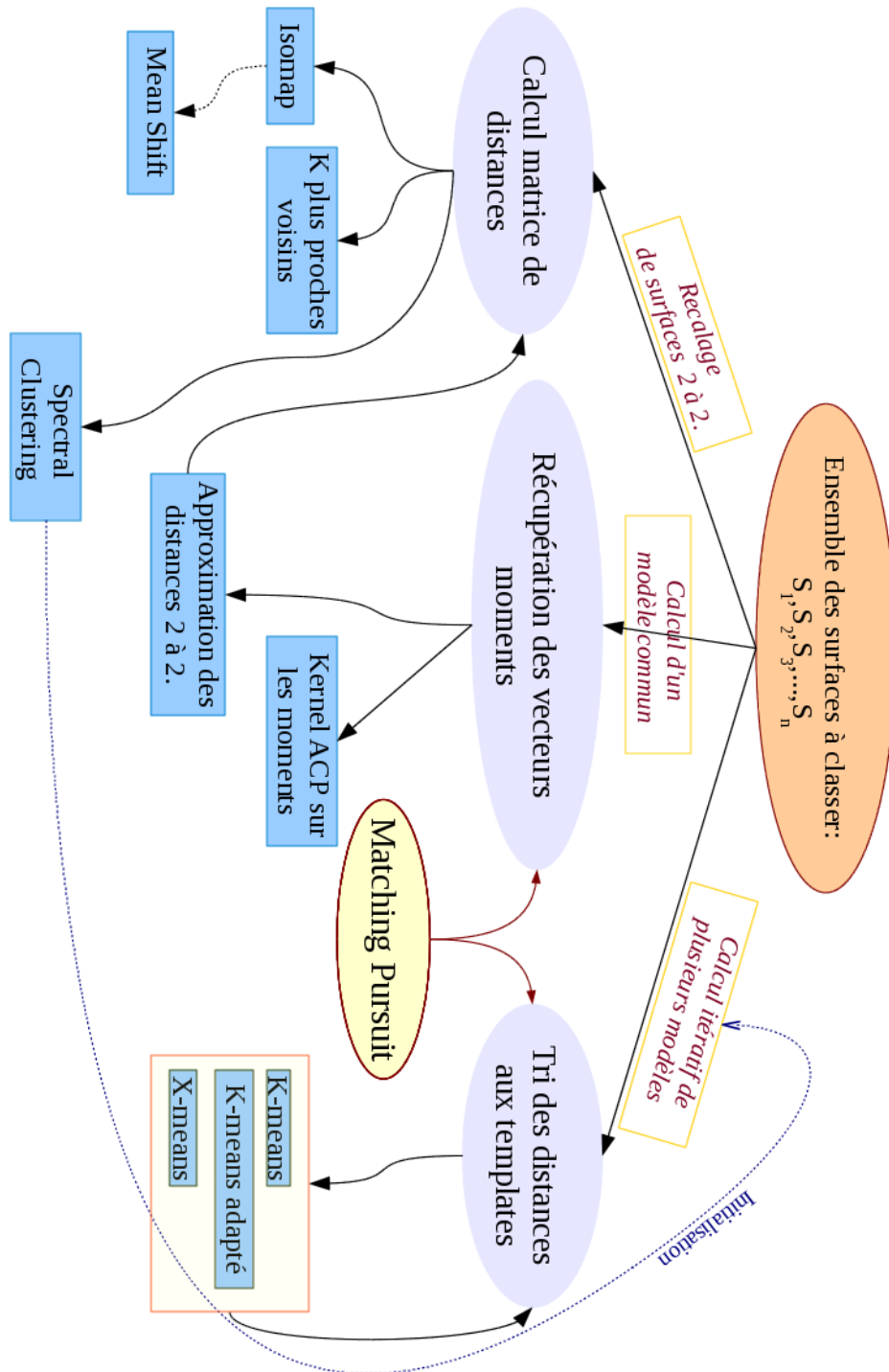
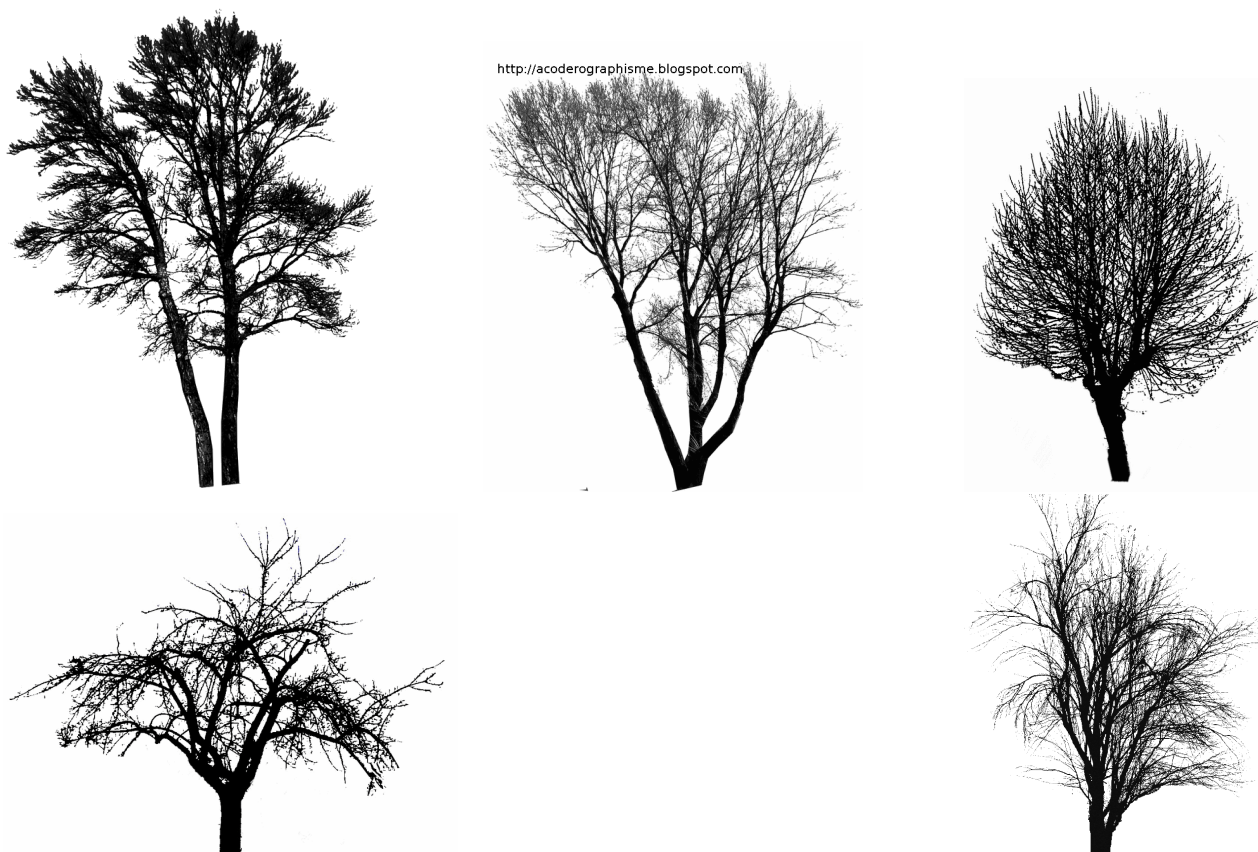


FIG. 4: Graphique récapitulatif des différentes approches étudiées durant le stage. La première classe d'approche (à droite) vise à estimer itérativement plusieurs modèles à partir de la population. La seconde classe (au milieu) repose sur l'analyse des déformations de chaque individu depuis un modèle unique. La dernière (à gauche) repose sur des recalages deux à deux entre les individus. Les pointillés correspondent à des perspectives qui n'ont pas encore été explorées.

2 Motivations et état de l'art

2.1 Notion de variabilité anatomique

On appelle variabilité anatomique le fait qu'au sein d'une même population, les structures changent de forme de manière significative d'un individu à l'autre. Cette variabilité inter-individuelle peut être issue d'un ou plusieurs modes de variation différents. Comme nous l'avons dit brièvement dans l'introduction, l'analyse multi modèle est intéressante lorsque l'on veut représenter une population issue de plusieurs modes de variation. Prenons l'exemple des arbres, leurs branches présentent des formes différentes, on ne retrouve jamais exactement la même morphologie d'un individu à l'autre et il existe une très grande variabilité inter-individu. Il serait logique, pour représenter la variabilité des arbres, de construire plusieurs modèles afin de résumer les différentes variations de la population ; un seul modèle ne serait pas représentatif.



Les hippocampes à étudier, dont la variabilité est importante viennent d'images IRM à 7 Tesla. Les IRM standard sont à 1,5 ou 3 Tesla ce qui permet une résolution spatiale dans le plan comprise entre 1 et 2 millimètres. A titre indicatif, le champ magnétique de la Terre est compris entre 30 et 50 μ Tesla. Les IRM à 7 Tesla sont dites à très haut champ, et permettent une résolution spatiale d'environ 0,25mm dans le plan, on a donc la possibilité d'observer beaucoup plus de détails anatomiques. L'apparition de ces nouveaux détails, que l'on ne pouvait voir qu'en histologie, à permis de remarquer que la tête de l'hippocampe était constituée d'un certain nombre, plus ou moins important, de « bosses » que l'on appelle des digitations. La construction d'un seul modèle à ce niveau de détail poserait des problèmes de représentation.

Un modèle est une sorte de moyenne ou représentant qui se veut représentative de la classe de laquelle il est issu. Un atlas est soit un modèle dans le cas où la population comprend une classe, soit plusieurs modèles.

2.2 Recalage d'images cérébrales

Etudier les similarités et différences entre les formes de deux sujets nécessite de les placer dans un référentiel commun. Cette étape s'appelle le recalage inter-individuel et consiste à rechercher la transformation permettant d'aligner au mieux les différents sujets de la population. Le problème du recalage d'images cérébrale a beaucoup été étudié. Les premières approches proposées reposent sur des transformations linéaires [9]. Ce type de déformation compense très mal la variabilité inter-individuelle (comme on peut le voir dans le cadre des sillons à la figure 8 sur la ligne du haut). C'est pourquoi des approches non linéaires ont été développées. La principale difficulté liée à ces approches est le contrôle de la régularité de la déformation, car des transformations trop irrégulières produiraient des absurdités anatomiques. Un cadre de déformation non linéaire particulièrement adapté est celui des difféomorphismes (transformation régulière, inversible, et d'inverse régulière). Parmi les approches difféomorphiques, le cadre des grandes déformations (LDDDM - Large Deformation Diffeomorphic Metric Mapping, voir chapitre 3) est le plus répandu. Ce stage ne portant pas sur les différentes méthodes de recalage, nous n'en parlerons pas plus ici et nous nous concentrons sur les constructions de modèles et d'atlas.

2.3 Une peu d'histoire : Talairach, MNI et ICBM

L'une des premières approches pour la création d'atlas du cerveau a été proposée par Talairach et Tournoux dans le cadre de la neurochirurgie en 1988 [30]. La normalisation spatiale de Talairach consiste en une déformation affine par morceaux qui amène tous les cerveaux dans un référentiel commun défini par un cerveau modèle, appelé l'atlas de Talairach, sur lequel ont été segmentées diverses structures.

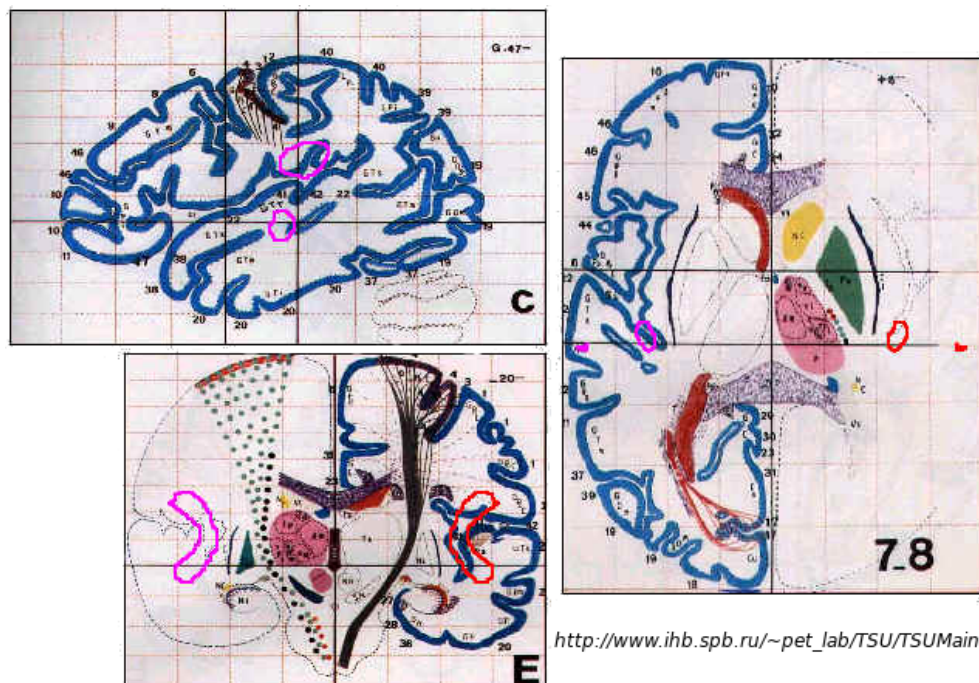
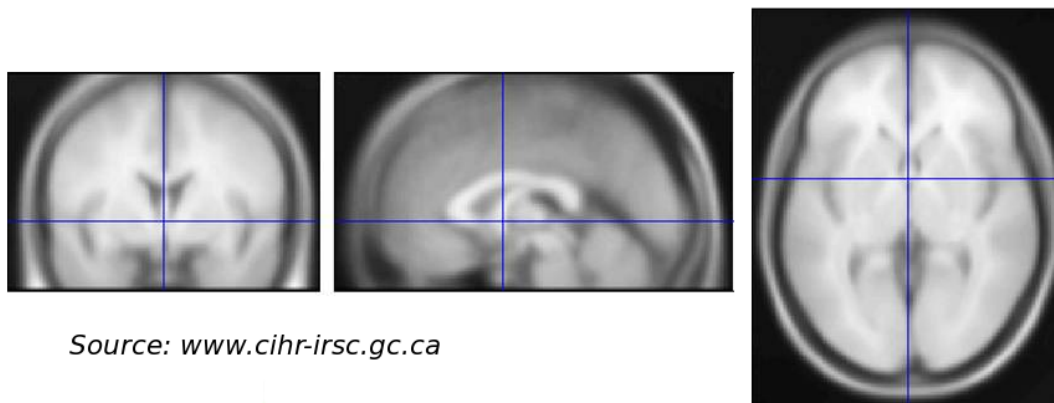


FIG. 5: De haut en bas et de gauche à droite : coupe sagittale, coupe coronale, et coupe axiale issues de l'atlas de Talairach.

Cet atlas a été durant plusieurs années le référentiel standard dans lequel de nombreuses études ont rapporté leurs résultats, malgré de fortes limitations inhérentes à la construction d'un tel atlas :

- Premièrement, l'atlas de Talairach provient de l'étude détaillée du cerveau d'un unique individu (et même d'un seul hémisphère) âgé de 60 ans. Il ne peut donc clairement pas être représentatif d'une population.
- Deuxièmement, cet atlas se présente sous la forme d'un schéma sur papier (figure 5) reproduisant des coupes espacées de 4mm alors que les IRM anatomiques sont des objets volumiques dont la résolution spatiale est de l'ordre du millimètre cube.

Ces limitations ont été levées au Montreal Neurological Institute (MNI) [13], qui a proposé un atlas probabiliste appelé MNI305 reposant sur 305 sujets jeunes (239 hommes et 66 femmes), ce qui est quand même plus représentatif de la variabilité anatomique. La technique proposée est entièrement automatique et repose sur un recalage affine global basé sur l'intensité des voxels des IRM.



Source: www.cihar-irsc.gc.ca

FIG. 6: Atlas standard MNI350 obtenu par transformation affine d'images IRM de 350 sujets.

L'atlas le plus utilisé aujourd'hui a été réalisé dans le cadre de l'International Consortium for Brain Mapping (ICBM) [24] et [25] à partir de 152 sujets, est appelé ICBM152. Les sujets ont été recalés sur l'atlas MNI305 par une transformation affine globale utilisant 9 paramètres.

Les atlas probabilistes du type ICBM et MNI qui normalisent tous les individus sont pratiques lorsqu'on ne cherche pas à identifier ou étudier des structures complexes comme les sillons ou petites comme l'hippocampe. L'utilisation de tels atlas serait complètement biaisée pour l'étude de l'hippocampe, structure qui n'est pas orientée de la même façon chez tout le monde. De plus ces atlas sont construits à partir d'une population prédéfinie, et ne sont donc pas représentatifs d'une population à étudier (par exemple une population d'âge différent ou de sujets atteints d'une pathologie). Un atlas qui s'adapte à la population serait alors plus représentatif.

2.4 Méthodes de construction d'un atlas adaptatif

Les méthodes adaptatives permettent de construire un atlas à partir de la population en même temps que le recalage, quelle que soit la population. La cible vers laquelle les individus sont recalés est donc raffinée de façon itérative. Au lieu d'utiliser un atlas déjà existant comme les atlas MNI, ICBM et Talairach, on construit un atlas adapté à l'étude d'une population donnée, par exemple des sujets atteints d'une certaine pathologie.

Lorenzen et al. en 2005 [20] ont proposé ce genre de méthode dans le cas d'images à niveau de gris. Ils utilisent la théorie des grandes déformations difféomorphiques [18] pour générer les transformations qui produiront l'atlas.

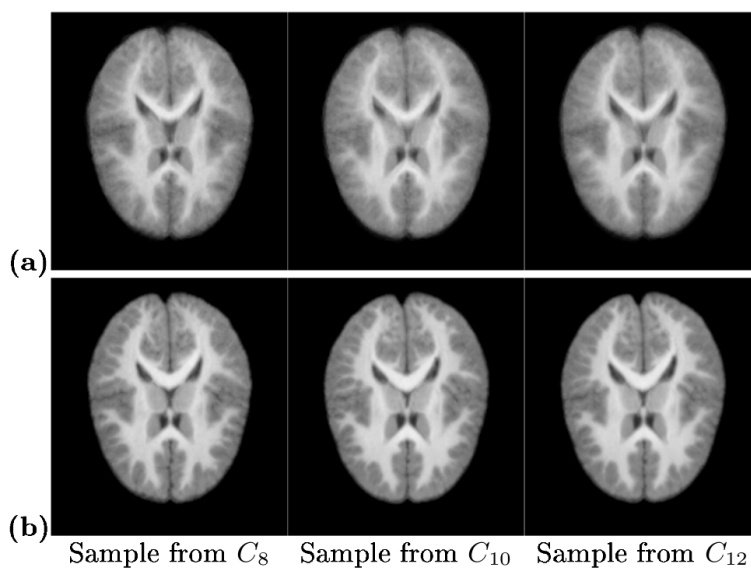


FIG. 7: *d'après [20]. En (a), moyenne de 8, 10 puis 12 cerveaux après un recalage linéaire. En (b), recalages par la méthode des grandes déformations diffeomorphiques à partir de 8, 10 et 12 cerveaux.*

Glaunès et al. en 2008 [16] ont proposé une méthode calculant un atlas adaptatif dont les transformations sont aussi basées sur la théorie des grandes déformations diffeomorphiques, mais cette fois dans le cas de courbes et surfaces et non plus d'images à niveaux de gris. La représentation des courbes et des surfaces peut se faire à partir de mesures ou de courants (voir chapitre 3). Il est ainsi possible d'obtenir une forme « moyenne » de surfaces ou de courbes. Nous verrons plus en détails cette méthode dans la partie suivante, car c'est cette méthode qui sera utilisée par la suite.

Dans le cas des sillons, Auzias a proposé durant sa thèse une méthode de recalage de sillons corticaux appelée DISCO [3] pour Diffeomorphic Sulcal-based Cortical registration. Comme dans les deux méthodes ci-dessus, les sillons sont recalés les uns sur les autres itérativement, pour construire un modèle adaptatif. La figure 8 compare le recalage linéaire des sillons corticaux au recalage diffeomorphique.

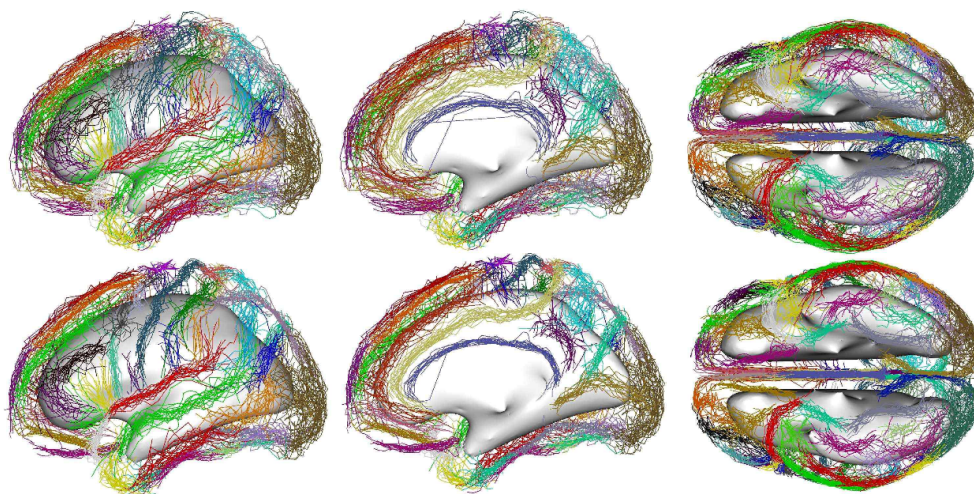


FIG. 8: *D'après [2]. La ligne du haut représente le recalage de sillons de 20 sujets par une méthode linéaire, en bas par la méthode DISCO.*

On voit que les atlas adaptatifs représentent beaucoup mieux la population, mais ne représentent

toujours pas plusieurs modes de variations. Pour cela, la création de plusieurs modèles est nécessaire.

2.5 Méthodes de modèles multiples

Présentons succinctement quelques méthodes de modèles multiples.

En 2007 Blezek et al. [4] présentent une méthode appelée « Atlas Stratification » qui, à partir d'une population, évalue si un ou plusieurs modèles sont nécessaires à la représentation de la population. Si plusieurs modèles sont requis, ils sont contruits à partir de sous ensembles de la population. Ils se basent sur l'algorithme du *mean shift* pour déterminer les modes [7] d'une population de manière itérative. Ensuite chaque mode de la population sera représenté par un modèle créé à partir des individus issus de ces modes.

En 2008, Allasonnière et al. [1] proposent une méthode pour déterminer différents modes au sein d'une population d'images à niveau de gris. L'approche repose sur des méthodes déterministes et stochastiques d'estimation de maximum a posteriori de modèles statistiques déformables. Mais elle n'a pas encore été adaptée au cas de courbes et de surfaces.

2.6 Méthodes continues d'analyse de la variabilité

A l'inverse des méthodes qui cherchent à estimer directement plusieurs modes, il y a des méthodes qui envisagent la variabilité de façon continue. Cette analyse continue peut être soit une fin en soi, soit une première étape à la détection de modes de variation.

La méthode bien connue qu'est l'Analyse en Composantes Principales (ACP) peut être utilisée pour analyser de façon continue la variabilité d'une population. Wang et al. s'y sont intéressés en 2007 [32] pour l'étude des hippocampes dans le cadre de la maladie d'Alzheimer en utilisant le cadre des grandes déformations. Et plus récemment Ceyhan et al. [6], développent une méthode qui utilise aussi les grandes déformations difféomorphiques pour analyser la morphologie de l'hippocampe et éventuellement détecter des différences entre des groupes ayant un diagnostic différent ou des changements de formes au cours du temps. Ces méthodes sont développées pour l'enveloppe externe de l'hippocampe segmenté sur des images IRM à 3 Tesla. Nous voulons nous intéresser à la forme de la structure interne de l'hippocampe segmenté sur des IRM 7T.

Une autre classe d'approches repose sur des méthodes d'apprentissage de variétés, comme ISOMAP (introduit par Tenenbaum et al. en 2000 [31]). Cet algorithme a la particularité d'utiliser des distances géodésiques entre les points (contrairement à l'algorithme Multi-Dimensional Scaling ou MDS, qui utilise des distances euclidiennes). Mais cette distance estimée par le plus court chemin d'un point à l'autre dans le graphe issu de la matrice de similarité initiale est « topologiquement instable » : des connexions erronées peuvent être créées dans le graphe. Certaines méthodes comme celle de Van der Maaten et al. en 2009 [22] suggèrent de supprimer les voisins d'un sommet qui violent la linéarité locale du graphe. Au court de sa thèse soutenue en 2011 [29], Sun a utilisé cette méthode pour analyser les sillons [28], et en particulier le sillon central, et en a extrait des résultats intéressants sur la latéralité gauchers/droitiers (voir figure 9 : la « bosse » qui correspond à la motricité de la main est plus ou moins centrée selon que l'on est gaucher ou droitier. L'ISOMAP est donc une technique intéressante si l'on cherche à expliquer des variations continues. Nous allons l'utiliser par la suite.

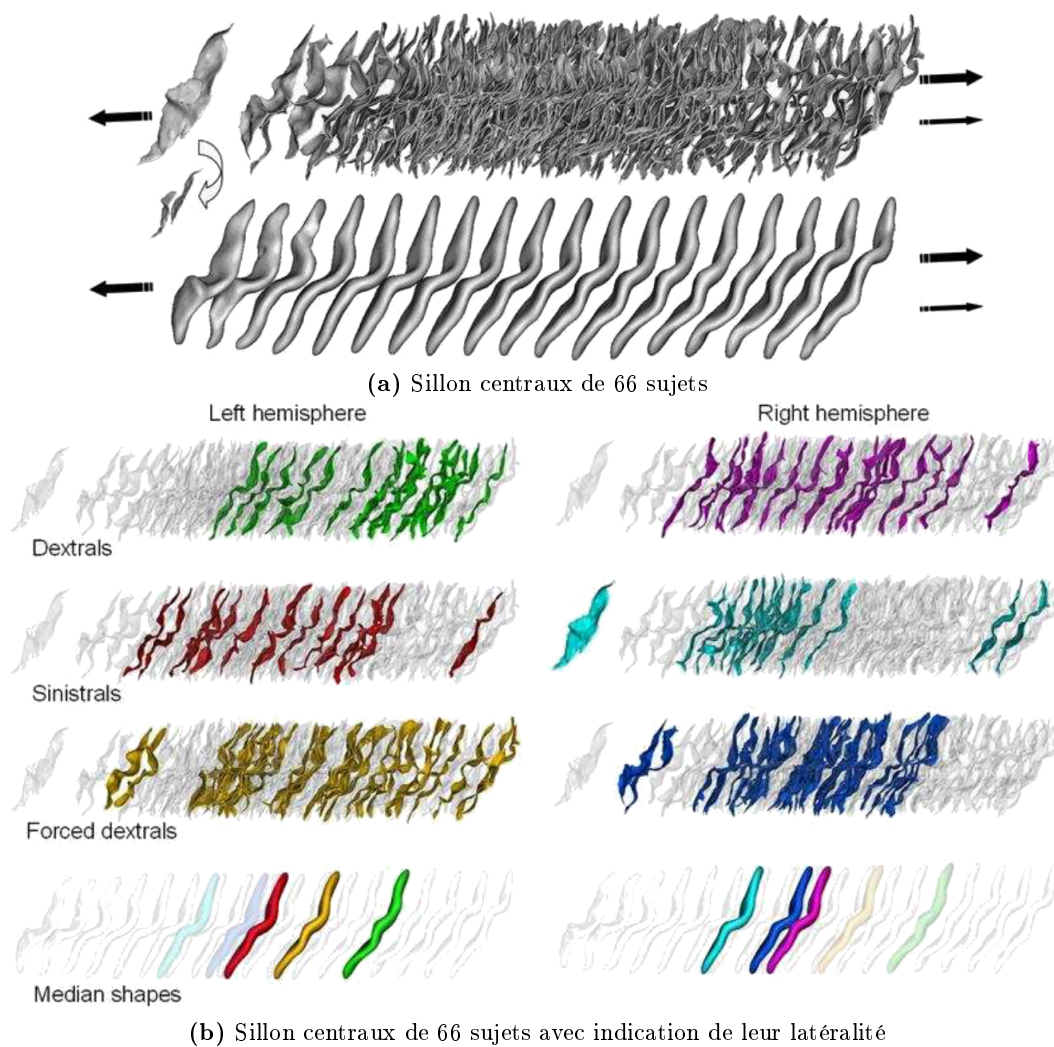


FIG. 9: D'après [29]. Localisation des populations droitier, gaucher et gaucher contrarié le long de l'axe ISOMAP. Code couleur : Cyan : gaucher, hémisphère droit ; Bleu : gaucher contrarié, hémisphère droit ; Violet : droitier, hémisphère droit ; Rouge : gaucher, hémisphère gauche ; Jaune : gaucher contrarié, hémisphère gauche ; Vert : droitier, hémisphère gauche

3 Modèles de grandes déformations et recalage de surfaces

Nous avons beaucoup parlé dans l'état de l'art de méthodes utilisant les difféomorphismes, qui sont utilisées pour recaler des courbes ou surfaces (ou images, mais pas dans notre cas) de façon non linéaire et très régulière.

Le modèle des grandes déformations a été construit à partir de la théorie fondatrice des splines de Bookstein [5] qui a initialement proposé d'écrire la transformation $\psi(x) = x + v(x)$ où v est le champ de déplacement. Ces déformations ne sont difféomorphiques que si le déplacement v reste petit, c'est-à-dire si ψ reste proche de la transformation identité. La régularisation du modèle LDDMM (Large Deformation Diffeomorphic Metric Mapping) permet d'obtenir une transformation correspondant à de grands déplacements tout en restant difféomorphe, en composant suffisamment (en fait une infinité dans le modèle mathématique) de petits déplacements $\psi_i(x) = x + v_i(x)$.

Nous allons dans cette partie présenter les outils et les notations mathématiques nécessaires à la compréhension des méthodes. Puis nous présenterons la méthode conduisant à la création d'un modèle unique représentant la population [15] qui sera par la suite utilisée.

3.1 Espaces de Hilbert à Noyaux Reproductifs : R.K.H.S.

Rappelons rapidement quelques notions élémentaires :

Définition 3.1 (Produit scalaire)

Un produit scalaire sur un espace vectoriel V sur \mathbb{R} est une application bilinéaire sur V notée : $\langle \cdot, \cdot \rangle_V : V \times V \rightarrow \mathbb{R}$ vérifiant les trois points suivant :

- Symétrie : $\forall v, w \in V, \langle v, w \rangle_V = \langle w, v \rangle_V$
- Positivité : $\forall v \in V, \langle v, v \rangle_V \geq 0$
- Défini : $\langle v, v \rangle_V = 0 \iff v = 0$

La norme induite par le produit scalaire est définie par $\|v\|_V = \sqrt{\langle v, v \rangle_V}$

Définition 3.2 (Espace de Hilbert)

Soit V un espace vectoriel sur \mathbb{R} . V est un espace de Hilbert si il est muni d'un produit scalaire complet pour la norme induite.

On rappelle qu'un espace métrique est complet si toutes suites de Cauchy $((x_n)_{n \in \mathbb{N}}$ telle que, $\forall \epsilon > 0, (\exists N \in \mathbb{N}) (\forall p, q \in \mathbb{N}) (p, q > N \Rightarrow \|x_p - x_q\|_V < \epsilon)$) convergent.

Les espaces hilbertiens sont en fait une généralisation des espaces euclidiens dans lesquels la dimension n'est pas forcément finie. Par exemple une fonction de \mathbb{R} dans \mathbb{R} est définie par l'ensemble des valeurs qu'elle prend, soit une infinité de nombres, et peut donc être vue comme un point dans un espace de dimension infinie.

Définition 3.3 (Espace dual)

Soit V un espace de Hilbert, on note $\langle \cdot, \cdot \rangle_V$ son produit scalaire et $\|\cdot\|_V$ sa norme.

On note V^* l'espace dual, l'espace qui contient toutes les formes linéaires continues $\mu : V \rightarrow \mathbb{R}$, de norme $\|\mu\|_{V^*} = \sup \{ \mu(f), \|f\|_V \leq 1 \}$

Présentons maintenant le théorème de Riesz qui permet d'établir l'existence d'une isométrie entre V et son espace dual V^* .

Théorème 3.1 (Théorème de représentation de Riesz)

Soient V un espace de Hilbert muni de son produit scalaire $\langle \cdot, \cdot \rangle_V$, et $\mu \in V^*$ une forme linéaire continue sur V .

Alors il existe un unique $\hat{\mu}$ dans V tel que : $\forall x \in V$ on ait $\mu(x) = \langle x, \hat{\mu} \rangle_V$

Définition 3.4 (Noyau)

On appelle noyau positif sur un ensemble S une fonction $K : S \times S \rightarrow \mathbb{R}$ telle que :

- Pour tout entier $n \in \mathbb{N}$, tous scalaires $\alpha_1, \dots, \alpha_n \in \mathbb{R}$ et tous éléments $x_1, \dots, x_n \in S$,

$$\sum_{i,j=1}^n \alpha_i \alpha_j K(x_i, x_j) \geq 0$$

Définition 3.5 (Espace de Hilbert à noyaux reproduisants)

Cette définition découle directement du théorème précédent.

Soit V un espace de Hilbert formé de fonctions sur un ensemble S . V est un espace à noyau reproduisant (R.K.H.S.) si les fonctionnelles d'évaluation $\delta_x : f \rightarrow f(x)$ sont continues pour tout $x \in S$

Soit la forme linéaire $\delta_x(f) = f(x)$. Si pour tout x , δ_x est continue, alors $\exists \hat{\delta}_x$ tel que $f(x) = \delta_x(f) = \langle f, \hat{\delta}_x \rangle_V$

$$\hat{\delta}_x(y) = K_V(x, y)$$

est appelé noyau reproduisant de l'espace V . On a alors : $\delta_x(f) = \langle f, K_V(x, \cdot) \rangle_V$.

Une interpolation par noyau (ou spline) est équivalente à une interpolation par minimisation d'une norme. En effet le noyau K_V de l'espace hilbertien V permet de résoudre le problème d'interpolation suivant :

$$\text{Minimiser } \|f\|_V \text{ tel que } f(y_i) = t_i \quad \forall 1 \leq i \leq n.$$

Ce qui nous intéresse, puisque nous cherchons à minimiser des distances. La solution de ce problème est donnée par :

$$f^*(x) = \sum_{j=1}^n \alpha_j K_V(x, y_j),$$

où les α_j sont solution du système linéaire :

$$\sum_{j=1}^n \alpha_j K_V(y_i, y_j) = t_i.$$

Ces vecteurs α_j seront appelés plus tard vecteurs moments et les champs de vecteurs optimaux sont des sommes de champs splines : $v_t(x) = \sum K_V(x_i(t), x) \alpha_i(t)$, avec $x_i(t)$ la trajectoire du point x_i au temps t .

3.2 Grandes déformations difféomorphiques

Comme nous l'avons dit un peu plus haut, la régularisation des grandes déformations difféomorphiques permet d'obtenir une transformation correspondant à de grands déplacements tout en restant difféomorphique : intuitivement, en composant une infinité de petits déplacements $\psi_i(x) = x + v_i(x)$, on construit la transformation ϕ par composition :

$$\phi = \psi_n \circ \psi_{n-1} \dots \circ \phi_1$$

de n difféomorphismes (déformations infinitésimales proches de l'identité). ϕ est alors un difféomorphisme qui peut être loin de l'identité et est défini comme une solution de l'équation différentielle :

$$\begin{cases} \frac{d\phi(t, x)}{dt} = v(t, \phi(t, x)), t \in [0, 1] \\ \phi(0, x) = x \end{cases}$$

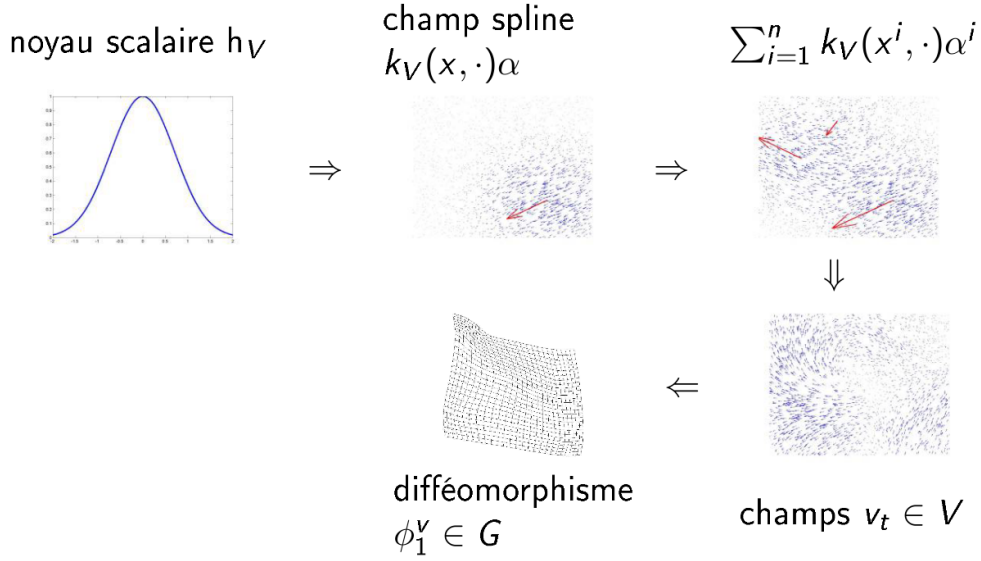


FIG. 10: Schéma illustrant les étapes permettant de passer de la définition du noyau au difféomorphisme, extrait de la présentation de soutenance de thèse de J. Glaunes [14]

avec $v(t, x)$ le vecteur vitesse du point x au temps t , et $\phi(t, x)$ la position de x au temps t

Définition 3.6 (Difféomorphisme)

C'est un modèle inspiré de la mécanique des fluides. Les difféomorphismes sont obtenus par intégration d'une famille de champ de vecteurs $(v(t, \cdot))$ d'un espace hilbertien V suffisamment régulier, telle que $\forall t \in [0, 1]$ on ait $\int_0^1 |v(t, \cdot)|_V dt < \infty$ selon l'équation de flot :

$$\frac{d\phi(t, x)}{dt} = v(t, \phi(t, x)), t \in [0, 1]$$

avec $\phi = \phi_1^v$ et $\phi_0 = Id$; la déformation commence au temps 0 et se termine au temps $t = 1$. Autrement dit, un difféomorphisme ϕ est donné par la formule suivante :

$$\forall z \in \mathbb{R}^3, \phi_t^v(z) = z + \int_0^t v_s \circ \phi_s^v(z) ds \quad (1)$$

On définit le groupe des difféomorphismes comme suit :

$$A_V = \{\phi_1^v \text{ tel que } v \in V\}$$

Le groupe est complet pour la métrique :

$$D_V^2(Id, \phi) = \inf \{\|v\|_V | v \in V, \phi_1^v = \phi\} \quad (2)$$

avec invariance à droite : $D_V^2(\phi, \psi) = D_V^2(Id, \psi \circ \phi^{-1})$

Cette distance évalue la quantité de déformation dans tout l'espace 3D.

3.3 Courants

L'idée des courants est de caractériser les formes (une forme peut être une courbe ou une surface) par leur action sur les champs de vecteurs.

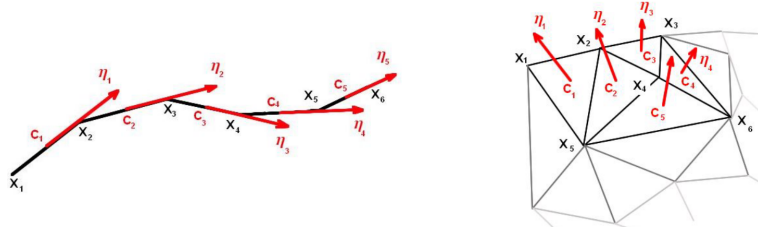


FIG. 11: A gauche discrétisation d'une courbe, et à droite discrétisation d'une surface. Dessin fait par J. Glaunès.

Définition 3.7 (Courants) Soient ω un champ de vecteurs et une forme S . On définit le flux du champ de vecteurs ω sur la surface S par l'équation :

$$S(\omega) = \int_S \omega(x)^t (u \wedge v)(x) d\sigma(x) \quad (3)$$

Avec $(u \wedge v)(x)$ la normale normalisée de la forme S au point x , et $d\sigma$ la mesure de Lebesgue sur la surface. On caractérise la forme S en mesurant la variation de son flux quand le champ de vecteurs ω varie.

Pour une courbe l'équation de flux par un champ de vecteurs est comme suit :

$$C(\omega) = \int_C \omega(x)^t \tau(x) d(x) \quad (4)$$

avec τ la tangente normalisée au point x de la courbe.

En discrétisant la forme S (voir figure 11), le courant de S est alors une somme de Dirac notée

$$[S] = \sum_{f=1}^{nf} \eta(f) \delta_{c(f)},$$

où f est le segment entre deux points consécutifs ou une face d'un maillage, $c(f)$ en est le centre, et $\eta(f)$ en est la normale ou la tangente.

Définition 3.8 (espace des champs de vecteurs) Soit K_W un noyau sur \mathbb{R}^3 et W l'espace hilbertien de noyau K_W . W est un espace de champs de vecteurs engendré par les champs de vecteurs de carré intégrable qui sont de la forme :

$$\omega(x) = K^W(x, y)\beta \quad (5)$$

pour tous points y et tous vecteurs β fixés. Le produit scalaire sur W est caractérisé par :

$$\langle K^W(., x)\alpha, K^W(., y)\beta \rangle_W = \alpha^t K^W(x, y)\beta \quad (6)$$

$$\Rightarrow \langle K^W(., x)\alpha, \omega \rangle_W = \alpha^t \omega(x) \quad (7)$$

On reconnait ici la propriété reproduisante des R.K.H.S.

L'espace des courants est l'espace des objets qui intègrent des champs de vecteurs.

Définition 3.9 (espace des courants) On note W^* l'espace des courants : W^* est l'espace des applications linéaires et continues de l'espace des champs de vecteurs W dans \mathbb{R} , ce qui en fait le dual de W , d'où la notation.

Il existe une application linéaire isométrique $L_W : W \rightarrow W^*$ puisque W est munit d'un produit scalaire. On a pour tout ω dans W :

$$L_W(\omega)(\omega') = \langle \omega, \omega' \rangle_W \quad (8)$$

$L_W(\omega)$ est un courant, c'est à dire une application de W dans \mathbb{R} .

On dit que $L_W(\omega)$ est la représentation duale du champ de vecteurs ω .

La représentation duale de la base vectorielle $K^W(x, \cdot)\alpha$ est $L_W(K^W(x, \cdot)\alpha)$, noté $\delta_x^\alpha : \delta_x^\alpha(\omega) = \langle \omega(x), \alpha \rangle$ pour tout ω

3.4 Estimation d'un template à partir d'une population [15]

Cette méthode proposée par Glaunès et al. [15] calcule un modèle commun à une population de surfaces ou de courbes représentés par des points non étiquetés. Nous nous intéressons ici au calcul d'un modèle à partir d'une population de surfaces ou de courbes représentées par des courants.

Soient N surfaces S_i dans \mathbb{R}^3 . On note $[S_i] = \sum_{f=1}^{nf} \eta(f)\delta_{c(f)}$ les courants correspondant aux S_i comme défini plus haut. La méthode consiste donc à trouver la surface « commune » aux S_i . Pour cela, toutes les surfaces vont être déformées afin que leurs déformations difféomorphiques soient les plus proches possibles les unes des autres, l'idéal étant qu'elles se superposent toutes. Ces déformations doivent donc coûter le moins cher possible tout en étant les plus proches les unes des autres. Il y a donc deux termes à minimiser pour trouver les N déformations optimales $\hat{\phi}_i$ conduisant au modèle $[\hat{S}]$: la distance entre les déformations et la taille de chaque déformation :

$$\left\{ \hat{\phi}_i, [\hat{S}] \right\} = \underset{\phi_i, [S]}{\operatorname{argmin}} \sum_{i=1}^N \left\{ \|[S] - \phi_i[S_i]\|_K^2 + \gamma D_V^2(Id, \phi_i) \right\}. \quad (9)$$

A ϕ_i fixés, le courant minimiseur $[S]$ de l'équation 9 est la moyenne des $\phi_i[S_i]$:

$$[\hat{S}] = \frac{1}{N} \sum_{i=1}^N \phi_i[S_i].$$

Cette moyenne ne correspond pas à une seule surface qui moyennerait la forme des autres surfaces, mais c'est l'union des surfaces déformées $\phi_i(S_i)$ pondérée par $1/N$. La formule de l'équation 9 peut être écrite comme suit, avec plus qu'un paramètre à minimiser, les difféomorphismes ϕ_i :

$$\left\{ \hat{\phi}_i \right\} = \underset{\phi_i}{\operatorname{argmin}} \sum_{i=1}^N \left\{ \left\| \left(\frac{1}{N} \sum_{i=1}^N \phi_i[S_i] \right) - \phi_i[S_i] \right\|_K^2 + \gamma D_V^2(Id, \phi_i) \right\}, \quad (10)$$

avec $D_V^2(Id, \phi_i)$ la distance de déformation du difféomorphisme ϕ_i défini plus haut, et γ un paramètre qui sert à donner plus ou moins de régularité à la déformation ; en effet si $\gamma = 0$ toutes les surfaces vont se superposer, mais elles seront toutes déformées. Dans le cas de surfaces représentant des structures anatomiques, il est important de garder ce paramètre non nul afin de ne pas trop déformer ces surfaces. Le premier terme de l'équation 10 est la mesure de dissimilarité induite par une norme d'un R.K.H.S pour un noyau de Cauchy K dont l'échelle est à définir par l'utilisateur : plus les données sont différentes, plus le noyau a intérêt à être grand. L'expression de la norme pour une somme vectorielle de masses de Dirac est donnée par :

$$\left\| \sum_{f=1}^{nf} \eta(f)\delta_{c(f)} \right\|_K^2 = \sum_{p=1}^{nf} \sum_{q=1}^{nf} \eta(p)^t K(c(p), c(q)) \eta(q) \quad (11)$$

4 Données d'hippocampes à très haute résolution spatiale

4.1 Segmentation

Les données d'hippocampes à très haute résolution sont celles de deux sujets sains décrits dans Chupin et al. Un troisième sujet avait été acquis mais n'a pas été utilisé dans le stage pour des raisons de temps, mais aussi pour cause de protocole d'acquisition différent des deux autres. Nous présentons brièvement ci-dessous les protocoles d'acquisition et de segmentation.

Les deux jeunes volontaires sains ont été scannés sur une IRM Siemens 7T (TrioTim), à l'université du Minnesota, avec une antenne de tête 16 canaux. L'acquisition a été réalisée avec une séquence *Turbo Spin Echo* pondérée en T2 avec les paramètres suivants TR/TE/flip-angle/Nex = 4000ms/77ms/60/1, acquisition matrix=512x512. La résolution dans le plan est de $0.375mm^2$, interpolée à $0.1875mm^2$, avec des coupes de 1.2mm et 1.1mm d'épaisseur avec un écart de 0.2mm et 0.3mm. La séquence a été répétée trois fois, les trois volumes ont été recalés et moyennés avec le logiciel SPM5. Le temps d'une acquisition est d'environ 8 minutes.

La segmentation manuelle a été réalisée par Marie Chupin (Cogimage) à l'aide du module de région d'intérêt du logiciel Anatomist de brainvisa (<http://www.brainvisa.info/index.html>), selon l'anatomie de l'hippocampe décrite dans Duvernoy [12]. Un protocole 3D cohérent a été utilisé qui permet de visualiser les différentes structures internes en 3D (stratum pyramidale de CA1-CA3, hilum du gyrus dentelé (gyrus dentelé proprement dit et CA4), subiculum, alveus et fimbria).

4.2 Extraction de squelette

Emilie Gerardin (thésarde de Cogimage, en collaboration avec le CMLA et Paris 5) a développé une méthode de modélisation de la corne d'Ammon sous la forme d'un squelette et d'une épaisseur. La corne d'Ammon est en effet un ruban de substance grise enroulé sur lui-même. Il est donc naturel de la modéliser comme une surface 2D sur laquelle on définirait une fonction d'épaisseur. L'approche proposée consiste à construire un champ de vecteurs v qui traverse le ruban de part en part dans le sens de la plus petite épaisseur. Les squelettes correspondant aux deux sujets sains sont présentés à la figure 12.

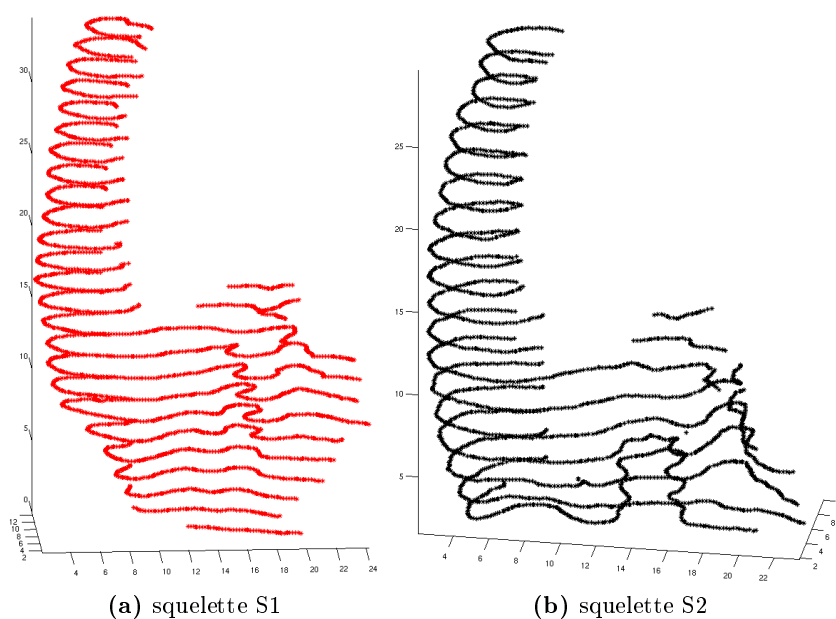


FIG. 12: Squelettes la corne d'Ammon pour deux sujets sains.

4.3 Les données synthétiques

Pour mettre au point et tester les méthodes, nous avons par ailleurs construit deux bases de données synthétiques. La première base est constituée de courbes gaussiennes. La seconde base est composée de squelettes d'hippocampes synthétiques construit par déformation des deux sujets sains décrits ci-dessus.

4.3.1 Courbes gaussiennes

Nous avons construit trois populations de courbes provenant d'une ou de deux gaussiennes dont les moyennes et les variances sont variables. Dans la population GA (figure 13), les courbes présentant deux gaussiennes ont des moyennes suffisamment espacées, alors que dans la population GB, certaines courbes ont des moyennes proches ce qui rend plus difficile la classification « une ou deux bosses ? » (voir figure 14). Ces deux populations de courbes ont été très utiles car elles permettent de tester assez rapidement les algorithmes, tout en connaissant le résultat. De plus, du fait de la simplicité des données, les erreurs (dues aux déformations) ont été moins difficiles à identifier.

Les deux populations ont été construites de la façon suivante. Les moyennes *moy* sont choisies aléatoirement parmi [12 13 13.5 14], et les écarts-types σ parmi [1 1.5 2]. On définit ensuite les trois familles de gaussiennes suivantes :

$$gauss1 = \frac{1}{\sigma \sqrt{2\pi}} * \exp(-((x - moy).^2)/(2 * \sigma^2)) \text{ qui servira pour les deux populations GA et GB}$$

$$gauss2A = \frac{1}{(\sigma + rand * 0.5) * \sqrt{2\pi}} \exp(-((x - (moy + 6 + rand * 8)).^2)/(2 * (\sigma + rand * 0.6)^2)) \text{ qui servira pour la population GA}$$

$$gauss2B = \frac{1}{(\sigma + rand * 0.5) * \sqrt{2\pi}} \exp(-((x - (moy + 4 + rand * 10)).^2)/(2 * (\sigma + rand * 0.6)^2)) \text{ qui servira pour la population GB.}$$

La population GA comprend 12 courbes à « une bosse » construites à partir de *gauss1* et 12 courbes à « deux bosses » construites à partir de *gauss1 + gauss2A*.

La population GB comprend 12 courbes à « une bosse » construites à partir de *gauss1* et 24 courbes à « deux bosses » construites à partir de *gauss1 + gauss2B*.

La population GC contient 3 classes : des « une bosse à gauche » faites avec *gauss1*, des « deux bosses » faites avec *gauss2A + gauss2B*, et des « une bosse à droite » issues de *gauss2B*.

4.3.2 Squelettes d'hippocampes synthétiques

N'ayant que trois vrais squelettes d'hippocampe, nous avons créé des populations d'hippocampes synthétiques à partir de déformations des deux squelettes réels. Les deux squelettes réels sont notés S1 et S2 et les différents types de déformations appliquées sont appelés A,B et C. L'application des différentes déformations A,B, C aux différents squelettes S1 et S2 produit des populations de squelettes synthétiques S1A, S1B, S2A, ...

Pour créer les squelettes synthétiques, nous avons défini des vecteurs moments initiaux aléatoires en chaque point des squelettes S1 ou S2, ainsi qu'un nombre *T* de pas le long de la trajectoire issue de ces vecteurs moments initiaux. Pour les populations de squelettes synthétiques de type A et B, un sujet sera créé pour chaque

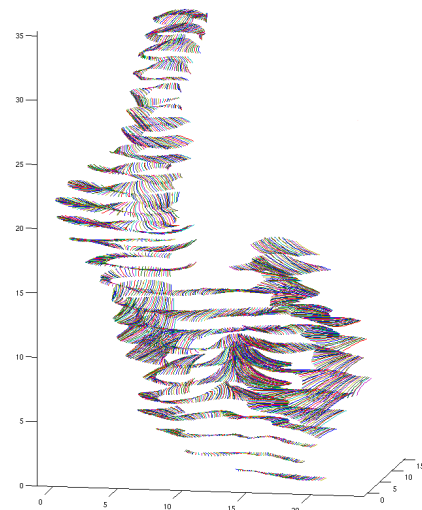


FIG. 16: trajectoires de la déformations du sujet S1 pour la construction de la population S1A décrite ci-après

temps t ($0 < t < T$) le long de la trajectoire. Les squelettes sont créés à l'aide d'une fonction de tir géodésique fournie par Joan Glaunès, qui permet de construire la trajectoire du tir géodésique effectué dans la direction des vecteurs moments initiaux, à partir du squelette, des vecteurs moments initiaux, du nombre de pas T et d'une taille de noyau σ qui définit le degré de régularité voulu.

On construit ensuite nos trois populations de type A, B et C. Les populations de type "A" sont plus simples que les "B" : les "A" comportent moins d'individus et ne sont pas trop déformées (comme la population GA : peu d'individus et bien séparés), les "B" au contraire contiennent plus de sujets et la déformation est plus importante. On voit que les sujets de la figure 18 sont bien plus différents les uns des autres que ceux de la figure 17. Les individus de la population de type "C" sont chacun issus d'une déformation aléatoire différente (contrairement à ceux des populations de type "A" et "B" qui correspondent aux différents pas de temps d'une la même déformation). La population S1C (que l'on peut voir à la figure 19) n'a donc pas de continuum. Les populations de type A sont construites avec $T = 5$ (ce qui produit 5 individus), $\sigma = 1.5$ et des vecteurs moments $0.5 * (rand(3, NbPoints) - 0.5)$, pour $NbPoints$ points en 3 dimensions. Cela produit 5 individus de chacun 3500 points pour la population S1A et 5 individus de 2750 points pour S2A

Les populations de type B sont construites avec $T = 15$ (ce qui produit 15 individus), $\sigma = 3$ et des vecteurs moments $0.7 * (rand(3, NbPoints) - 0.5)$, pour $NbPoints$ points en 3 dimensions. Cela produit 15 individus de chacun 3500 points pour la population S1B et 5 individus de 2750 points pour S2B

Les populations de type C sont construites avec $T = 2$, $\sigma = 2$ et des vecteurs moments $0.5 * (rand(3, NbPoints) - 0.5)$, pour $NbPoints$ points en 3 dimensions. Cela produit 5 individus de chacun 3500 points pour la population S1C et 5 individus de 2750 points pour S2C

Trois populations mixtes (mélangeant les populations issues des sujets S1 et S2) ont ainsi été créées : la population S1S2A, qui réunit les deux populations S1A et S2A. La population S1S2B qui réunit les deux populations S1B et S2B. La population S1S2A est plus facilement séparable, puisqu'elle est issue de deux classes de type "A" (comme pour la population A des courbes), qui sont des classes peu déformées de leur modèle original. Et enfin la population S1S2C issue de la réunion des deux classes S1C et S2C.

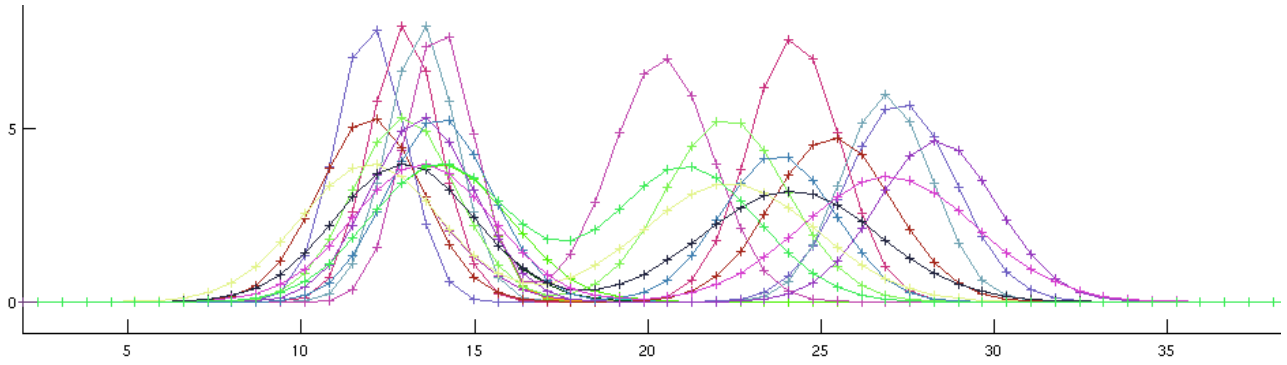


FIG. 13: Population GA : 24 courbes. Les 12 premières n'ont qu'une bosse et les 12 suivantes ont deux bosses

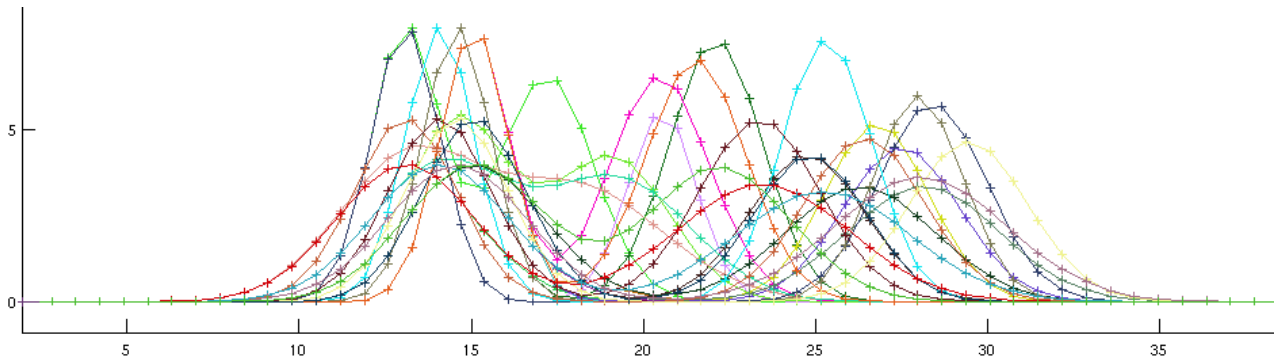


FIG. 14: Population GB : 36 courbes. Les 12 premières n'ont qu'une bosse et les 24 suivantes deux bosses. On peut observer que les courbes à « deux bosses » sont plus difficiles à distinguer de celles à « une bosse » que dans la population GA.

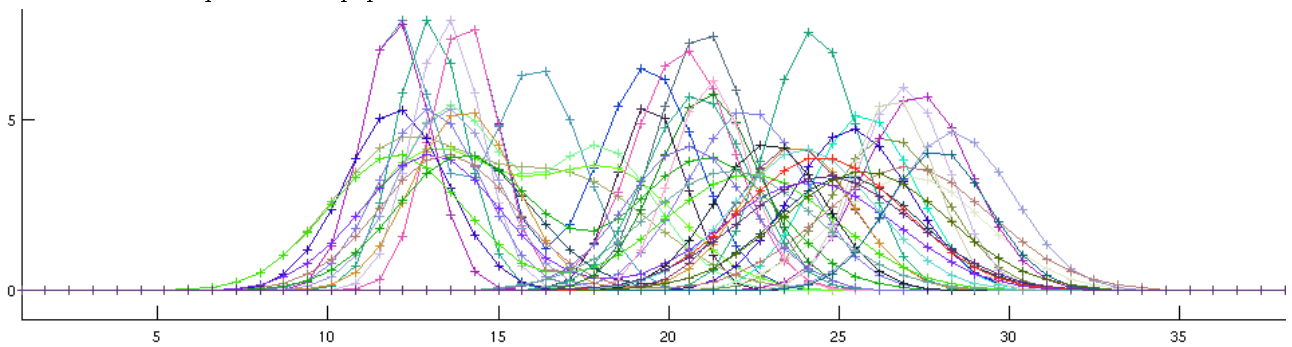


FIG. 15: Population GC : 48 courbes. Les 12 premières n'ont qu'une bosse et les 24 suivantes deux bosses, les 12 dernières ont une bosse.

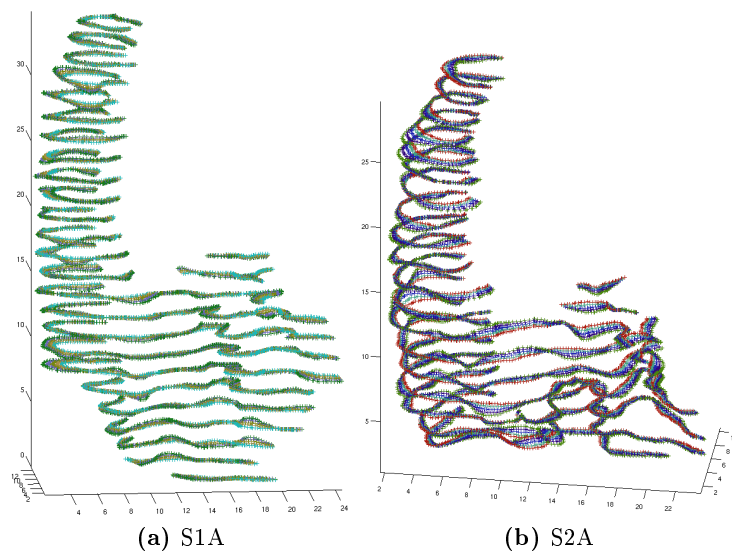


FIG. 17: Les populations d'hippocampes "A". Pour chaque population, une couleur représente un individu

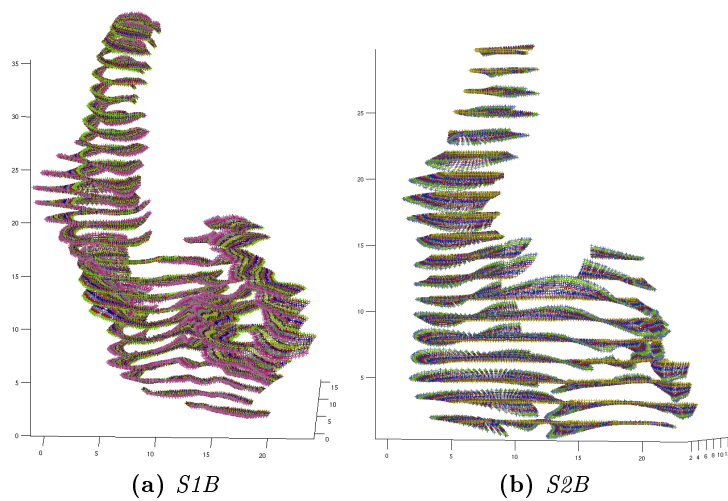


FIG. 18: Les populations d'hippocampes "B". Pour chaque population, une couleur représente un individu

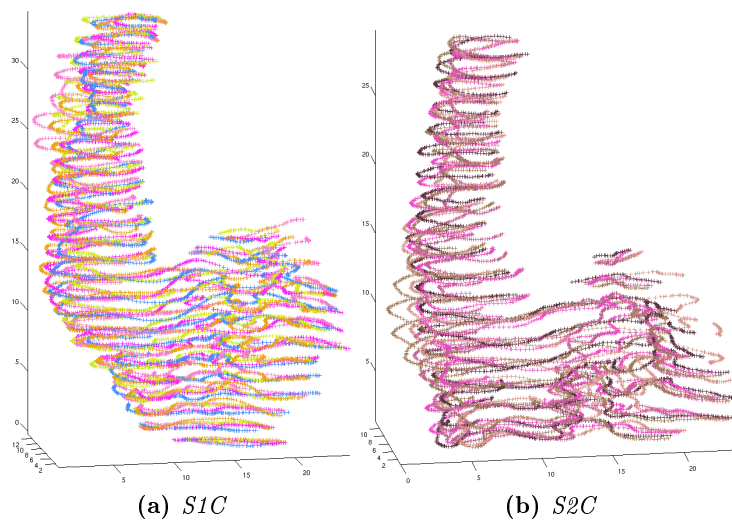


FIG. 19: Les populations d'hippocampes "C". Pour chaque population, une couleur représente un individu

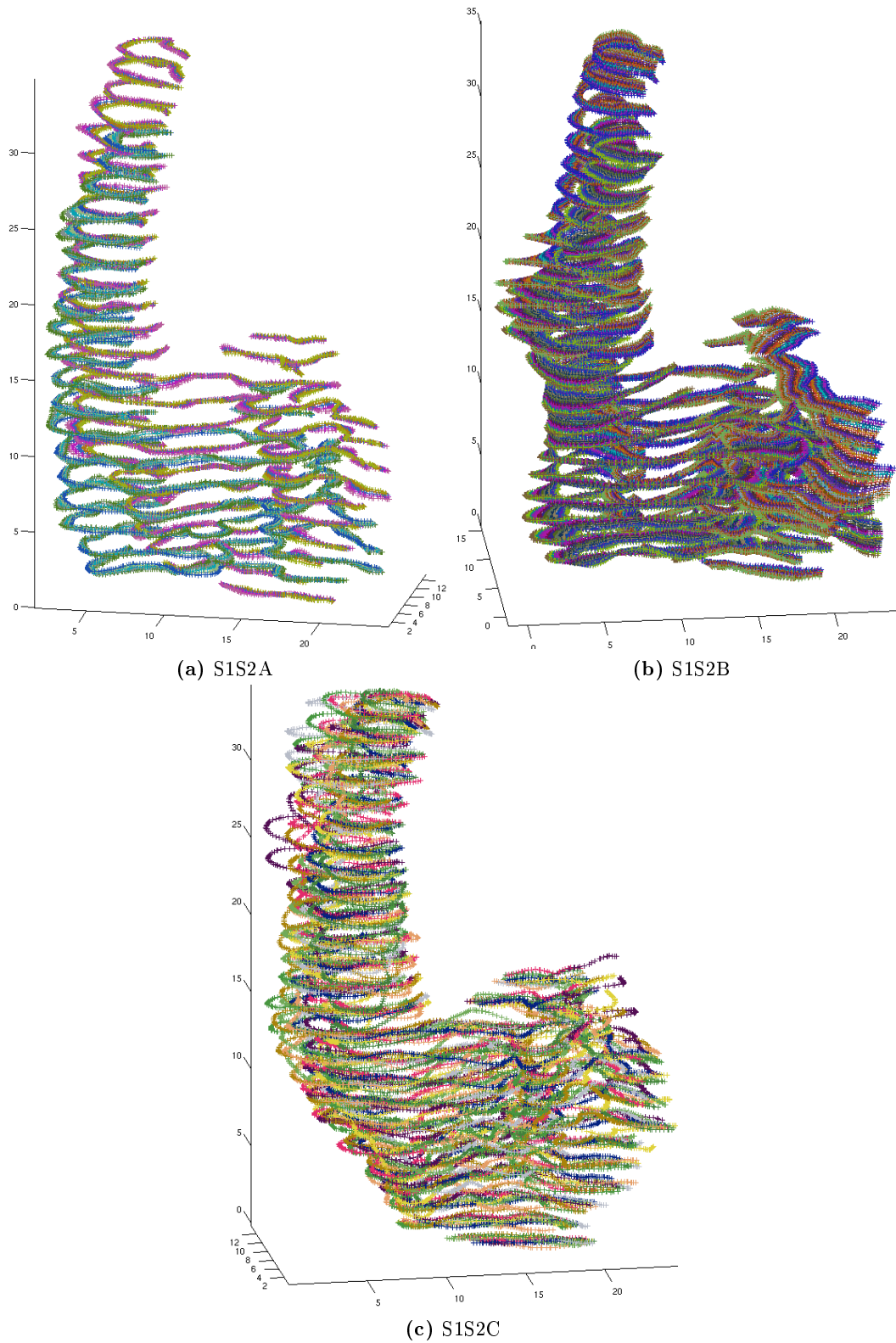


FIG. 20: Trois populations mixtes d'hippocampes. Chaque couleur représente un individu. (a) La population S1S2A contient 10 individus : les 5 de la classe S1A et les 5 de la classe S2A, soit 31250 points. (b) La population S1S2B en contient 30 : S1B + S2B, soit 93750 points. (c) La population S1S2C contient 10 individus : les 5 de la classe S1C et les 5 de la classe S2C, soit 31250 points.

5 Approche itérative de regroupement et de recalage simultané

L'approche que nous allons présenter maintenant vise à construire itérativement les différents modèles de la population. Cette partie correspond à la branche de droite de la figure 4.

5.1 Intuition

Comme nous l'avons vu précédemment, la méthode d'estimation d'un modèle à partir d'une population par transformations difféomorphiques proposée par Glaunès dans [17] et [15] est obtenue par l'équation suivante :

$$\arg \min_{\varphi_j, \mu} \sum_{j=1}^N \left(\|\varphi_j x_j - \mu\|_{K_V}^2 + \gamma D^2(Id, \varphi_j) \right), \quad (12)$$

avec x_j les N courbes ou surfaces de la population, $\varphi_j x_j$ leurs transformations et μ le modèle.

Or, nous voulons, à partir d'une population de courbes ou surfaces trouver n modèles. Rappelons en une equation la méthode des K-moyennes.

$$\arg \min_{\mathbf{S}} \sum_{i=1}^K \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \mu_i\|^2, \quad (13)$$

avec x_j les individus de la population et μ_i les centres des K ensembles de points S_i . Rappelons aussi

Algorithm 1 K-moyennes

Require: Un ensemble de points 2D (x_i) à classer en un nombre K de clusters.

- 1: Initialisation aléatoire des K centres $C(k)$
 - 2: **repeat**
 - 3: Affectation de chaque x_i au cluster dont il est le plus proche du centre $C(k)$
 - 4: Recalcul des centres : $\forall k \in \{1, \dots, K\}, C(k) =$ moyenne des x_i affectés à la classe k .
 - 5: **until** Les centres $C(k)$ ne bougent plus.
-

l'algorithme des K-moyennes, qui est utilisé pour partitionner les différents sujets d'une population (Algorithme 1).

On peut voir qu'il est facile d'adapter la formule 12 pour arriver à une formule du type de celle du K-moyennes (formule 13). La formule serait alors :

$$\arg \min_{\varphi_i, \mathbf{S}} \sum_{i=1}^K \sum_{\varphi_j x_j \in S_i} \left(\|\varphi_j x_j - \mu_i\|_{K_V}^2 + \gamma D^2(Id, \varphi_j) \right), \quad (14)$$

La distance de chaque individu j aux k "moyennes" μ_i n'est plus une distance euclidienne, mais sa distance de déformation vers le modèle μ_i : $D^2(Id, \varphi_j)$ qui est la distance entre cet individu j et sa transformation finale.

Les μ_i ne sont plus des moyennes comme dans la formule du K-moyennes, mais des modèles, comme dans l'équation 12 : c'est-à-dire, un modèle est issu de l'union des meilleurs transformations $\varphi_j x_j$ des courbes ou surfaces de l'ensemble S_i .

Le calcul des distances de chaque sujet (ou individu) aux modèles nécessite donc un recalage difféomorphique de chaque sujet sur chaque modèle, ce qui coûte assez cher en temps de calcul. Le calcul d'un modèle de 24 courbes comprenant chacune une soixantaine de points nécessite environ 3 minutes sur GPU pour 7 étapes de recalages (nécessaires pour obtenir un bon modèle). Le calcul des K-moyennes (par l'équation 14) pour ces mêmes 24 courbes, soit environ 1300 points (donc de très petites données) avec une seule étape de recalage pour la création du modèle à chaque itération dure environ 10 minutes sur GPU.

5.2 Le problème : les distances

Comme il fallait s'en douter, sur le papier tout a l'air plus simple et donne l'impression de fonctionner sans trop de peine, mais en pratique, appliquer simplement cette formule n'aboutit pas aux résultats voulus : dans le cas d'une "mauvaise" initialisation, les modèles formés des individus "autours" de ces centres n'évolueront pas correctement et n'auront aucun sens par rapport aux individus présent dans la population. Par exemple, plaçons nous dans le cas de deux classes U et V : si les deux centres choisis appartiennent à la même classe finale (disons que les deux centres appartiennent à la classe U), tous les individus de la classe U seront séparés et répartis dans les deux classes, car ils seront proches des deux centres choisis lors de l'initialisation. Il va se passer la même chose pour les individus de l'autre classe : ils seront loin des deux centres mais seront répartis dans les deux classes. Le modèle de chaque classe, qui est alors composé de l'union des déformations des individus qui lui ont été affecté, aura une forme plus ou moins intermédiaire entre les deux classes, ce qui n'amènera pas l'algorithme à converger normalement. En effet, à l'itération suivante, le centre de chaque classe n'est plus un seul individu, mais un ensemble d'individus, donc la déformation de chaque individu se fera sur un ensemble d'individus (déformés) constituant le modèle, les individus auront donc plus de "choix" pour satisfaire les contraintes de déformations, c'est-à-dire que l'individu ira se caler sur la partie du modèle hétérogène qui lui correspond le plus (déformation la moins couteuse qui amène le plus près possible de la cible). Pour éviter ceci, il faut augmenter le nombre d'étapes de recalages afin de construire un modèle assez fin, ce qui par ailleurs augmente considérablement le temps total d'exécution.

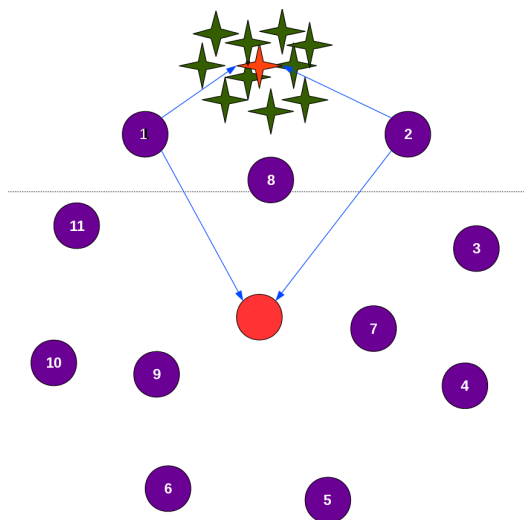


FIG. 21: Deux classes de surfaces. Le trait pointillé représente la séparation par la méthode des K-moyennes. Les traits pleins représentent la valeur de la distance de l'individu aux deux centres des classes colorés en rouge.

De plus, un individu se trouvant à une distance de 10 d'un modèle et d'une distance de 100 d'un autre ne veut pas forcément dire qu'il doit appartenir à ce premier, comme le montre la figure 21, la distance du rond "1" au modèle étoilé est plus petite que la distance de ce rond "1" au modèle de sa classe. Voyons comment nous pouvons parer à ce problème.

5.3 Quelques solutions

Afin de régler le problème des distances faussées du fait du trop grand nombre de points constituant la cible (c'est-à-dire le modèle), le recalage des individus vers les modèles a été changé : ce sont maintenant les modèles qui se déforment pour se caler sur les individus, ainsi la distance de déformation est

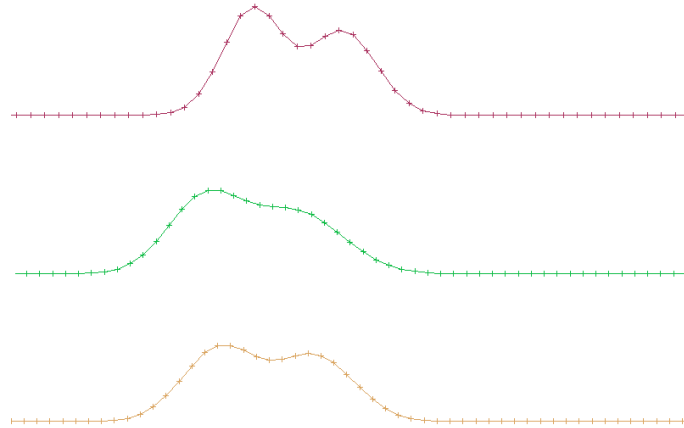


FIG. 22: Quelques courbes issues de la classe GB, n'ayant pas deux bosses séparées

un peu plus longue, d'autant plus que le modèle ne comporte pas beaucoup d'individus ressemblant à l'individu cible. Les sujets de la population sont maintenant les cibles. Pour les grandes populations, nous avons utilisé un Matching Pursuit [11] afin de réduire le nombre de points des modèles en ne gardant que les plus pertinents (voir 5.4) afin de réduire le temps de calcul.

Dans l'algorithme du K-moyennes, au moment du choix du cluster, les points sont triés simplement : si le point p est plus proche du cluster k , il est affecté à la classe k . C'est ce qui était fait dans la première version avant de se rendre compte que les distances n'étaient pas aussi bien distribuées. En effet, elles sont plutôt comme le montre la figure 21. Pour cela, nous avons donc choisi de ne s'intéresser qu'aux rangs des sujets par rapport aux modèles : le sujet se trouve à une distance de 100 d'un modèle tout en étant le 2^{me} plus proche de ce dernier, et à une distance de 10 de l'autre modèle en étant le dernier, donc le plus éloigné, alors ce sujet sera classé dans le premier modèle, celui dont il est distant de 100.

A la figure 21, par exemple, les points 1, 8 et 2 sont plus proches du modèle étoile, mais en sont plus loin que toutes les étoiles, ils n'appartiennent donc pas à la classe étoile.

Pour empêcher les mauvais classements, les individus ne sont pas tous classés. L'algorithme ne partitionne donc plus la population comme le fait le K-moyennes à l'origine, mais ne met dans ces classes que les individus les plus caractéristiques. En effet, dans notre cas, il n'est pas nécessaire de partitionner l'ensemble des individus, puisque nous voulons obtenir les différents modes de variation de la population, ainsi l'algorithme ne classe dans les clusters que les sujets étant "proche" d'un modèle et "loin" des autres, il est donc nécessaire de définir un seuil délimitant les individus proches et lointains, et une marge, afin qu'un individu proche de $seuil - 1$ d'un cluster et loin de $seuil + 1$ d'un autre ne soit pas classé, car il se trouve finalement entre les deux. Les biais liés aux sujets entre deux classes sont éliminés (par exemple, dans la classe GB , cf figures 14 et 22, les courbes ayant deux gaussiennes dont les deux moyennes sont suffisamment proches pour n'avoir ni une bosse, ni deux bosses séparées). Nous appellerons, afin de les différencier, l'algorithme du K-moyennes avec ce traitement des distances et l'inversion du recalage : K-moyennes-adapté.

De plus, pour éviter le problème des mauvaises initialisations, si l'une des classes est vide, ce qui grâce au seuil et à la marge est quelque chose de tout à fait possible et qui arrive lorsque par exemple le modèle d'au moins une des classes n'est représentatif d'aucun individu de la population, on réinitialise les centres des classes (cf. figure 28). Bien sûr cette condition suppose qu'il existe au moins deux classes et donc que la population n'est pas un continuum issu d'un même mode de variation. Une

Algorithm 2 K-moyennes adapté

Require: Tous les points de tous les sujets Suj à classer en K classes. un seuil $seuil \in \mathbb{N}$ et une $marge \in \mathbb{N}$

```

1: initialisation : On choisit aléatoirement  $K$  numeros  $init_k$  dans  $[1; nbSuj]$  pour initialiser les
   centres :  $C_k = [Suj(init_k)]$ 
2: for  $k = 1$  to  $K$  do
3:    $source_k = CreateSource(C_k)$  (cf section 3.4)
4: end for
5: for  $s = 1$  to  $nbSuj$  do
6:    $target_s = CreateTargetCourants(Suj(s))$  (cf section 3.4)
7: end for
8: while il y a du changement dans les classes do
9:   for  $k = 1$  to  $K$  do
10:    for  $s = 1$  to  $nbSuj$  do
11:     On recale les  $K$  cibles sur les  $nbSuj$  sujets pour en connaitre leurs distances :  $S_k =$ 
     registrationDiffo( $source_k, target_s$ )
12:      $Distance_k(s) = S_k.distIdPhi$  (voir équation 2 page 19)
13:    end for
14:     $D_k = triCroissant(Distance_k)$ 
15:   end for
16:   affectation des sujets aux classes :
17:    $sujClass = \emptyset$ 
   Pour chaque classe on examine les candidats
18:   for  $k = 1$  to  $K$  do
19:     $candidat_k = D_k(1 : seuil) - sujClass$ 
20:    for  $c = 1$  to  $length(candidat_k)$  do
21:     for  $k2 = 1$  to  $K$  and  $k2 \neq k$  do
22:       $ecart_c(k2) = [|candidat_k(c) - find(D_{k2} == candidat_k(c))|]$ 
23:      if  $ecart_c(k2) < marge$  then
24:        break ; On regarde le candidat suivant
25:      end if
26:     end for
27:     if  $k2 == K - 1$  then
28:        $C_k \leftarrow candidat_k(c)$ 
29:        $sujClass \leftarrow candidat_k(c)$ 
30:     end if
31:   end for
32: end while

```

Return : les listes C_k comptenant les indices des sujets classés

autre alternative a été envisagée : faire un Spectral Clustering (voir section 7.2) sur un échantillon de la population (pour de grandes populations) afin de mieux initialiser le K-moyennes-adapté, avec des sujets éloignés. Cette dernière solution n'a pu être testée lors du stage.

5.4 Matching Pursuit

Le Matching Pursuit est une méthode introduite par Mallat en 1993 [23], qui consiste à trouver une base qui correspond au mieux à une fonction f d'un espace hilbertien. Supposons qu'il existe une famille $(g_{\gamma_n}(t))_{\gamma_n \in \Gamma}$, telle que pour toutes fonctions $f(t)$ appartenant à un espace de Hilbert, il existe $(a_n)_{n \in \mathbb{Z}}$ telle que :

$$f(t) = \sum_{-\infty}^{+\infty} a_n g_{\gamma_n}(t) \quad (15)$$

Les a_n sont des coefficients. Le Matching Pursuit donne, après m itérations, la décomposition suivante du signal f précédent :

$$f = \sum_{n=0}^{N-1} \langle R^n f, g_{\gamma_n} \rangle g_{\gamma_n} + R^N f, \quad (16)$$

avec $R^n f$ le reste de f à l'itération n : $R^n f = R^{n-1} f - \langle R^{n-1} f, g_{\gamma_{n-1}} \rangle g_{\gamma_{n-1}}$.

L'erreur $\|R^n f\|$ décroît exponentiellement, et sert de condition d'arrêt à l'algorithme. Passé un certain seuil de précision, l'algorithme s'arrête.

Cette méthode de Matching Pursuit a été ensuite adaptée en 2009 au cas des courants par Durrleman et al. [11] pour trouver une estimation itérative de points et de vecteurs qui approximent au mieux le problème de déconvolution suivant :

$$T = L_W(\gamma) \quad (17)$$

Les notations sont les mêmes que dans la partie 3 : T est un courant, γ est un champ de vecteurs et L_W est l'opérateur linéaire permettant de passer de l'espace de champs de vecteur W à l'espace des courants W^* .

L'algorithme cherche donc, pour un champ de vecteurs $\gamma \in W$ donné (ainsi qu'un seuil de précision), les points et les vecteurs associés dans tout l'espace qui sont tels que le courant $T' = \sum_{i=1}^N \alpha_i \delta_{x_i}$ soit le plus proche de T défini précédemment. On veut en fait que T' soit la projection de T sur l'espace engendré par les δ_{x_i} , et donc que $T - T'$ soit orthogonal à cet espace, ce qui fait que l'on veut que (en se rappelant que $\delta_x^\alpha(w) = \langle K_W(x, \cdot) \alpha, w \rangle_W$) :

$$\forall \text{ points gardés } j \quad , \quad \left\langle L_W(\gamma) - \sum_{i=1}^N \alpha_i \delta_{x_i}, \delta_{x_j} \right\rangle_{W^*} = 0 \quad (18)$$

$$\iff \langle L_W(\gamma), \delta_{x_j} \rangle_{W^*} = \left\langle \sum_{i=1}^N \alpha_i \delta_{x_i}, \delta_{x_j} \right\rangle_{W^*}$$

$$\iff \langle \gamma, K_W(x_j, \cdot) \rangle_W = \left\langle \sum_{i=1}^N \alpha_i K_W(x_i, \cdot), K_W(x_j, \cdot) \right\rangle_W$$

$$\iff \delta_{x_j}(\gamma) = \delta_{x_j} \left(\sum_{i=1}^N \alpha_i K_W(x_i, \cdot) \right)$$

$$\iff \gamma(x_j) = \sum_{i=1}^N \alpha_i K_W(x_i, x_j) \quad (19)$$

Si l'on connaît les positions optimales des points (x_i) dans \mathbb{R}^3 , il ne reste plus qu'à déterminer les vecteurs α_i associés. On veut que :

$$\sum_{i=1}^N \alpha_i \delta_{x_i} = L_\omega(\gamma) \iff \sum_{i=1}^N L_\omega^{-1}(\alpha_i \delta_{x_i}) = \gamma \quad (20)$$

$$\iff \sum_{i=1}^N K_W(x_i, \cdot) \alpha_i = \gamma(\cdot) \quad (21)$$

L'équation 21 est un système linéaire et se résout facilement pour trouver les α_i . Par contre trouver les x_i est un problème NP-complet, comme l'a montré Davis en 1997 (dans [10]). La méthode de Durrleman et al. [11] propose donc de prendre comme premier point x_1 de l'espace \mathbb{R}^3 , le point qui explique la plus grande partie du signal (ici du champ de vecteurs γ , c'est-à-dire le point qui maximise la norme du champ de vecteurs $\|\gamma(x)\|_{\mathbb{R}^3}$). Une fois le premier point trouvé, le premier vecteur α_1 se trouve simplement à partir de l'équation 21 : $\alpha_1 = K_W(x_1, x_1)^{-1} \gamma(x_1)$. On calcule le champ de vecteurs résiduel en retirant du champ de vecteurs total le champ de vecteurs formé par les nouveaux points x_i et vecteurs α_i . Lors de la première itération, on ne retire donc que $K_w(x_1, \cdot) \alpha_1$, et ainsi de suite jusqu'à ce que la norme du champ de vecteurs résiduel passe sous un certain seuil passé en paramètre.

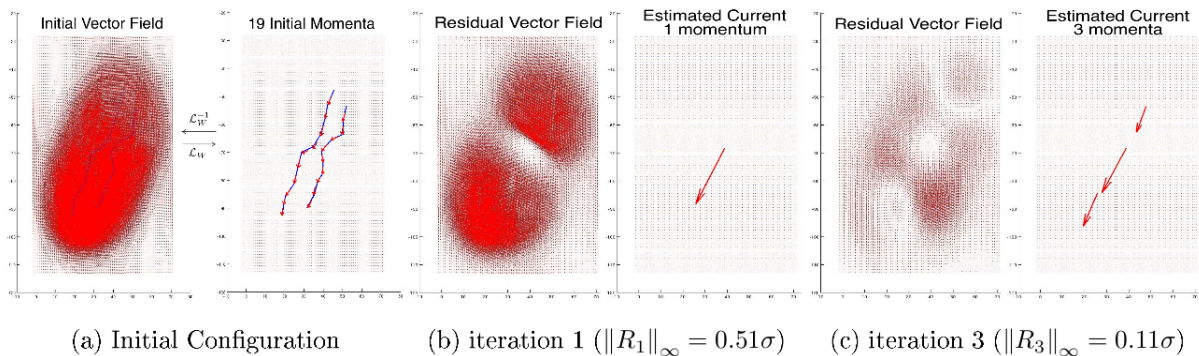


FIG. 23: D'après [11]. En (a) on voit à gauche le champ de vecteurs produit par les deux courbes bleues et à droite les vecteurs moments α en rouge. En (b) le premier point avec son vecteur moment qui maximise la norme du champ de vecteurs de (a), la partie gauche de (b) montre le champ de vecteurs résiduel, donc sans les vecteurs $L_W^{-1}(\alpha_1)$. L'image (c) montre la troisième itération, il y a donc 3 points et trois vecteurs moments ; le champ de vecteurs résiduel a encore réduit.

La taille du noyau K_W déterminée par son σ est aussi importante, car plus le noyau sera grand, moins de points seront gardés à précision égale, puisque le noyau "interdira" de garder des points trop rapprochés. De même un noyau trop petit autorisera la retenue de points proches, mais le champ de vecteurs résiduel diminuera moins rapidement, puisque l'influence de chaque point sur son entourage est diminuée par la taille du noyau. Il faut donc prendre une taille de noyau adaptée aux données.

Pour notre utilisation, nous avons légèrement modifié cet algorithme (voir algorithme 3) : nous ne cherchons pas les points parmi tout l'espace \mathbb{R}^3 , mais parmi les points de la surface (ou de la courbe), ce qui a l'avantage de ne pas avoir à positionner de grille. Ainsi nous pouvons faire tourner l'algorithme sur les vecteurs moments (qui sont nuls partout sauf sur les points de la forme), et non plus sur les champs de vecteurs. De plus le Matching Pursuit n'est pas exécuté sur une seule surface, mais sur un ensemble de surfaces, car nous voulons l'appliquer sur le modèle afin qu'il contienne moins de points. On ne cherche plus le point x_1 qui maximise le champ de vecteurs γ , mais le point x_1 qui maximise le champ de vecteur du sujet ayant le champ de vecteur maximum, car nous avons alors autant de champs de vecteurs que de sujets. Chaque sujet a son champs de vecteurs.

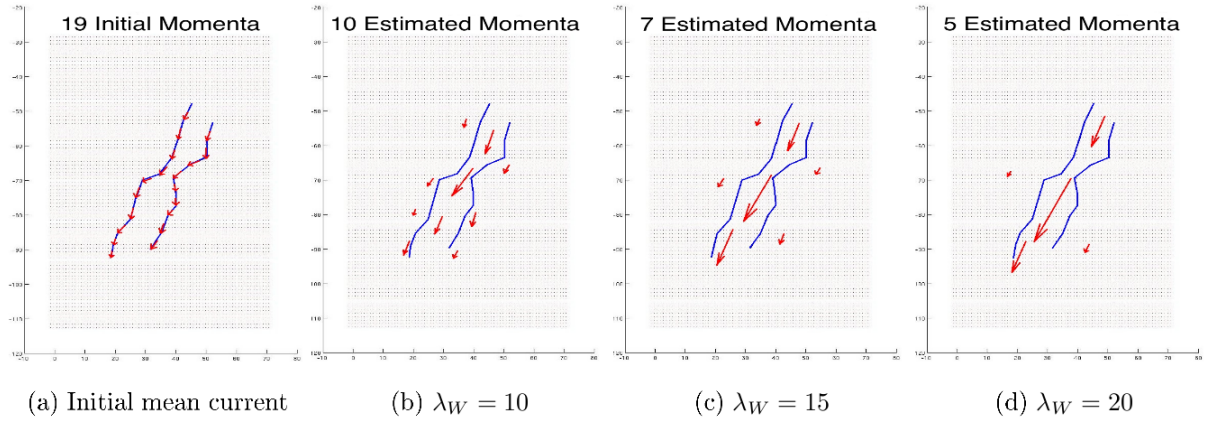


FIG. 24: D'après [11]. λ_W est pour nous le σ du noyau K_W . À précision égale, on remarque que la taille du noyau influence elle aussi le nombre de points

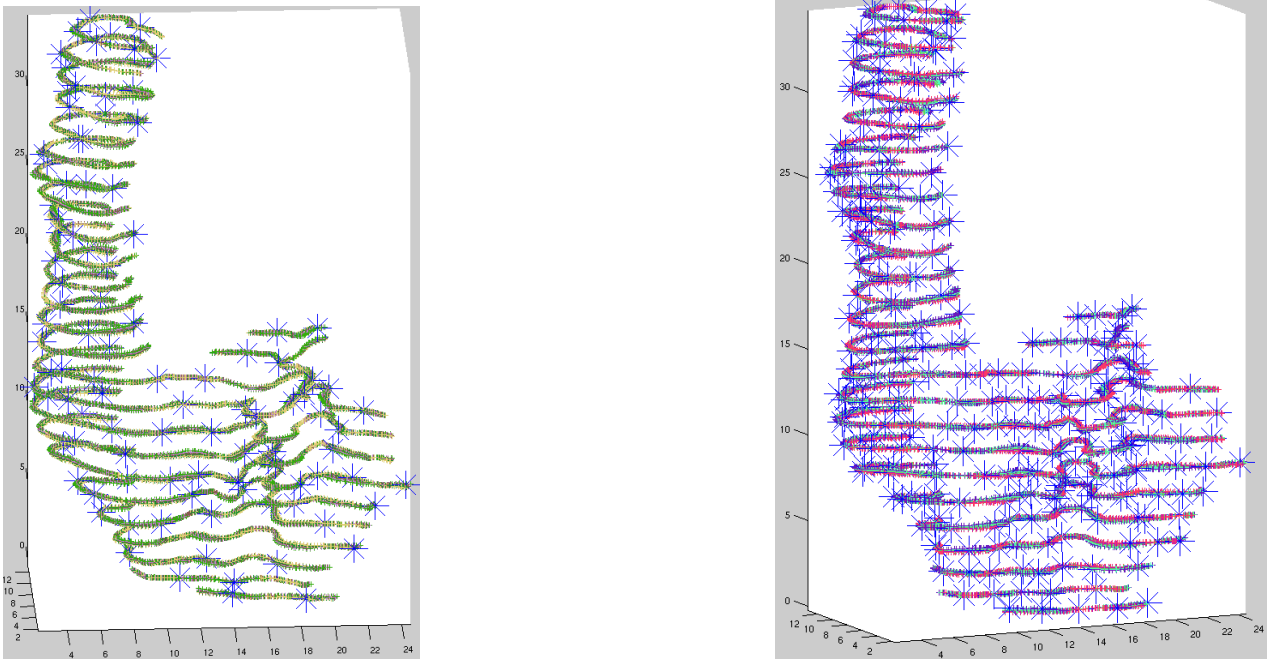


FIG. 25: Modèle d'une famille de squelettes d'hippocampes comprenant 17500 points. En bleu, les points gardés par le Matching Pursuit pour un σ choisi, à gauche une précision de 10^{-3} a permis de garder 100 points et à droite une précision de 10^{-6} a retenu 255 points

Les résultats du Matching Pursuit (MP) sont plutôt bons avec une taille de noyau K_W adaptée, même en ne choisissant les points que parmi ceux déjà existant, car pour un ensemble d'environ 2000 points, avec une précision de 10^{-11} , le Matching Pursuit conserve une quarantaine de points (cf figure 32). Le Matching Pursuit a été intégré dans la fonction de création d'un modèle afin d'accélérer son temps d'exécution. À chaque co-recalage, un MP est fait sur la cible qui contient donc tous les points de tous les sujets. Dans le cas d'algorithmes itératifs, le Matching Pursuit est fait sur le modèle courant de la classe pour réduire la taille des données. Il n'est par contre nécessaire que sur de grandes données.

Algorithm 3 Matching Pursuit

Require: Plusieurs champs de vecteurs : $\gamma(s) : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, le noyau K_W , l'ensemble des points du modèle x_{old} et un seuil $\eta \in \mathbb{R}$

- 1: $nbSujet = length(\gamma)$,
- 2: $\gamma_1(s) = \gamma(s)$ pour tout $s \in [1; nbSujets]$,
- 3: $N = 1$
- 4: **while** $\|\gamma_N\|_\infty / \|\gamma\|_\infty > \eta$ **do**
- 5: $ind_{suj} = argmax_{s \in [1; nbSujets]} \|\gamma_N(s)\|_\infty$
- 6: $ind = argmax_{x \in x_{old}} \|\gamma_N(ind_{suj})\|$
- 7: $x_N \leftarrow x_{old}(ind)$
- 8: **for** $s = 1$ to $nbSujet$ **do**
- 9: Calcul des vecteurs moments $\alpha^N(s)$ à l'étape N en resolvant :
- 10: $\sum_{p=1}^N (K(x_i, x_p) \alpha_p^N(s) = \gamma(s)(x_i)$ pour $i \in [1; N]$
- 11: On calcul le nouveau champs de vecteur résiduel :
- 12: $\gamma_{N+1}(s) = \gamma(s) - \sum_{i=1}^N K_W(\cdot, x_i) \alpha_i^N(s)$
- 13: $N \leftarrow N + 1$
- 14: **end for**
- 15: **end while**

Return : La liste des points gardés x_N avec leurs vecteurs moments α^N correspondant

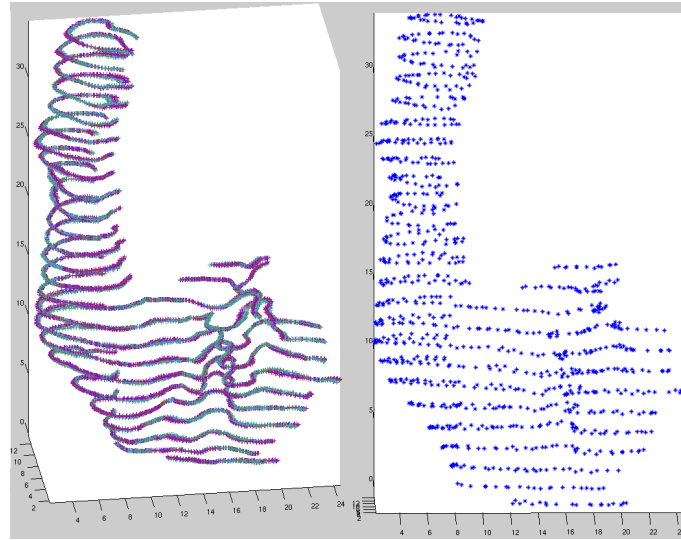


FIG. 26: Le même modèle à gauche (17500 points), à droite les 1043 points gardés par le MP avec une précision de 10^{-10}

5.5 Résultats

Les résultats sont de la figure 27 à la figure 30. Les résultats du K-moyennes sont présentés à la figure 27. On peut voir qu'une mauvaise initialisation (ici pour la classe 2) ne pourra pas être corrigée par la suite. Par ailleurs, même avec une initialisation correcte (ici pour la classe 1), l'utilisation des distances de déformation n'est pas adéquate. Ces deux problèmes sont corrigés par le K-moyennes adapté (figure 28). Sur la figure 29 on peut voir l'importance de la marge : sans marge les individus à la limite entre deux classes sont mal classés. La figure 30 présente les deux modèles obtenus à l'issue d'un K-moyennes adapté.

Remarque : dans les résultats, certaines initialisations sont faites avec une courbe, d'autres avec deux. L'idée de prendre deux courbes vient du fait que dans l'algorithme du K-moyennes l'initialisa-

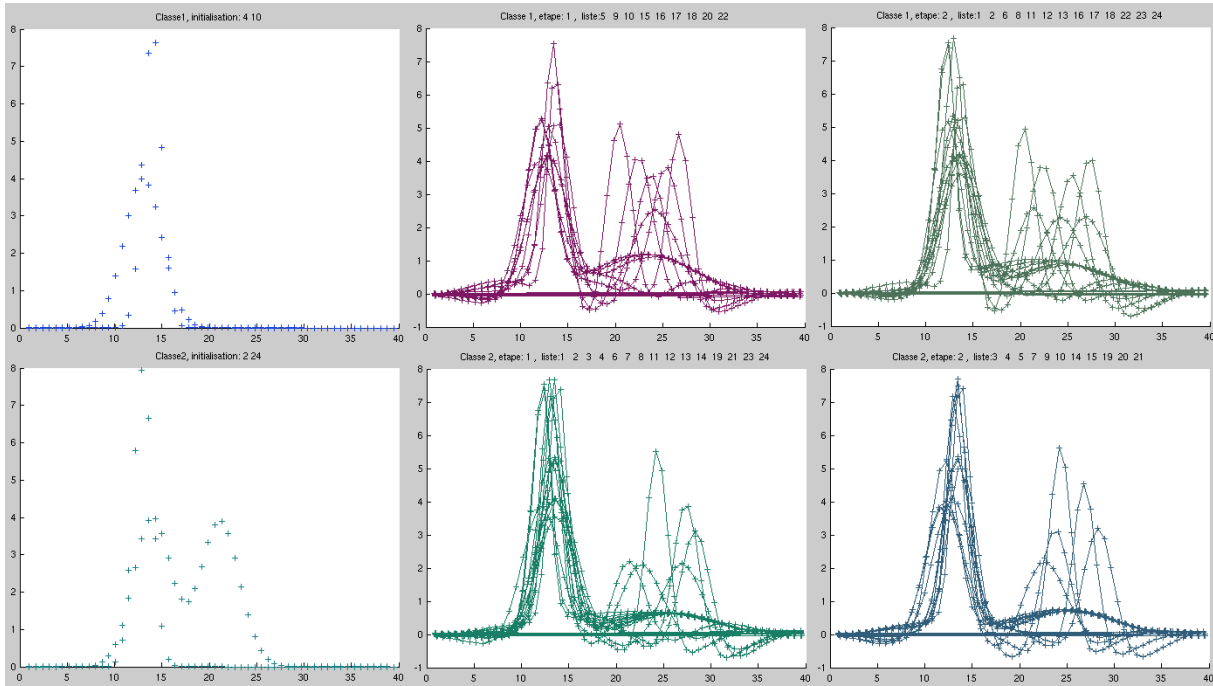


FIG. 27: Application de l'algorithme du K -moyennes sur la population GA . Ici et dans les figures suivantes, la ligne du haut représente la classe 1 et la ligne du bas la classe 2. A gauche : initialisation. Au centre : itération 1. A droite : itération 2. Ici on ne voit que les deux premières itérations, mais l'algorithme n'arrivera pas à bien classer les courbes. On peut constater deux problèmes. D'une part l'initialisation de la classe 2 n'est pas correcte. D'autre part, même avec une bonne initialisation comme dans la classe 1, l'utilisation directe de la distance de déformation ne conduit pas à une bonne classification.

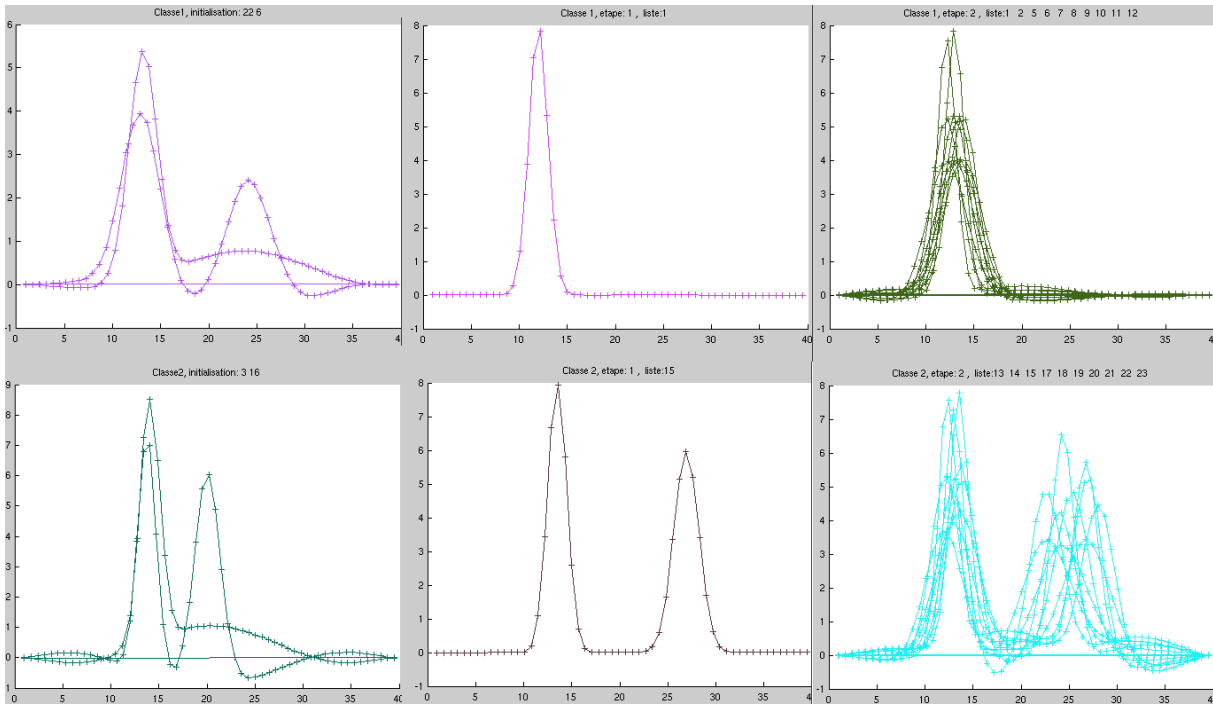


FIG. 28: K -moyennes-adapté sur la population GA , avec une première mauvaise initialisation, et une marge de 2 (différence de rang entre un classement proche et un classement loin). La mauvaise initialisation est tout de suite corrigée.

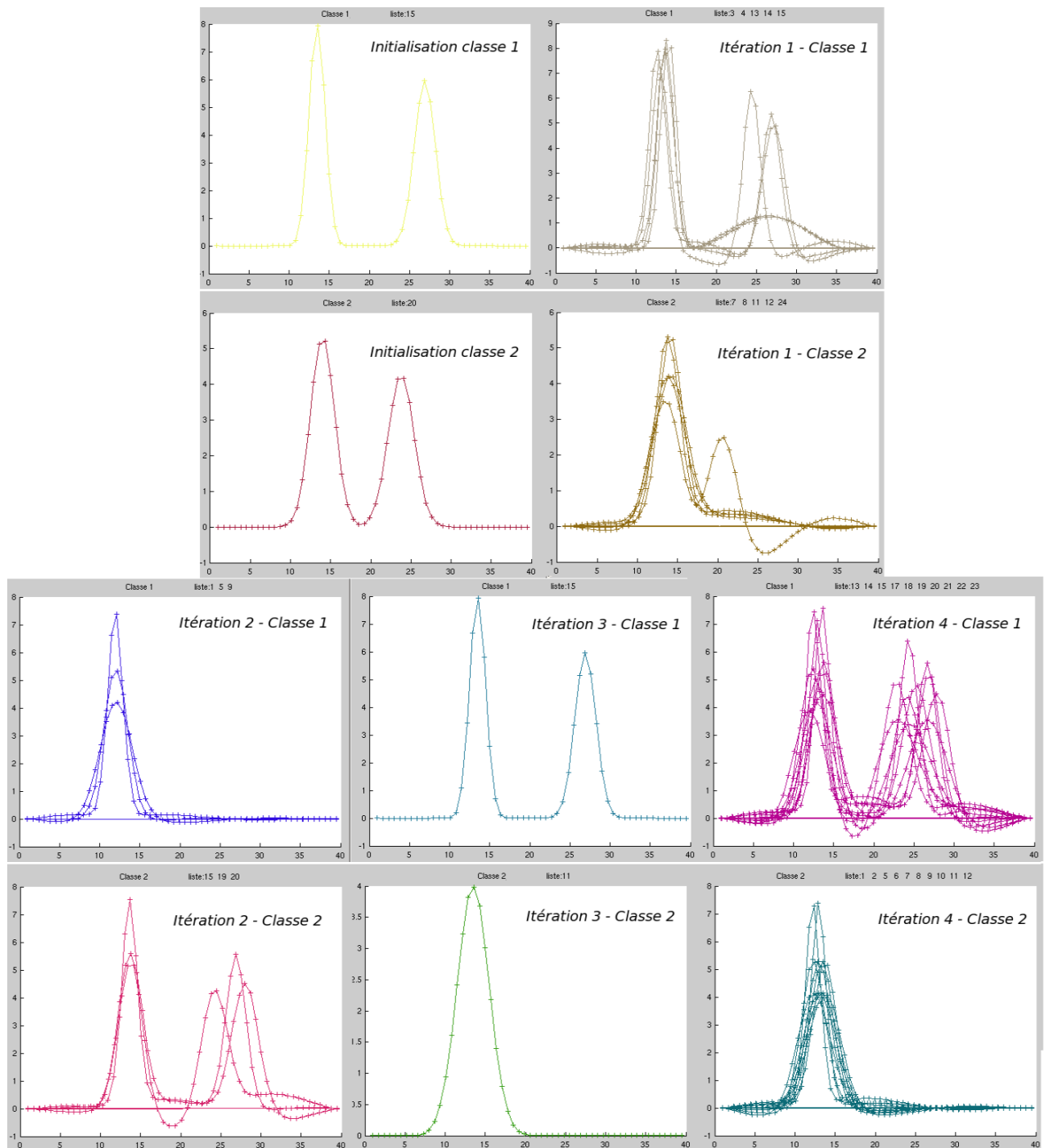


FIG. 29: *K*-moyennes-adapté avec une première mauvaise initialisation, sans marge. Du fait de l'absence de marge, la réinitialisation ne se fait pas tout de suite. L'itération 2 ne trouve que 3 courbes proches de la classe 1, du fait de l'amplitude de la première bosse (contre celle, plus petite, de la classe 2). La classe 2 trouve aussi quelques courbes, mais les deuxièmes bosses sont trop différentes et lors de la classification (pour l'itération 3), les courbes les plus proches de la classe 2 à l'itération 2 ne seront pas "loin" de la classe 1, car ce seront des courbes présentant plutôt une bosse, puisque la seule partie "claire" du modèle est la première bosse. Il y a donc réinitialisation à l'itération 3, qui tout de suite, à l'itération 4 classe bien une bonne partie des courbes, les 4 courbes manquantes sont entre les deux classes.

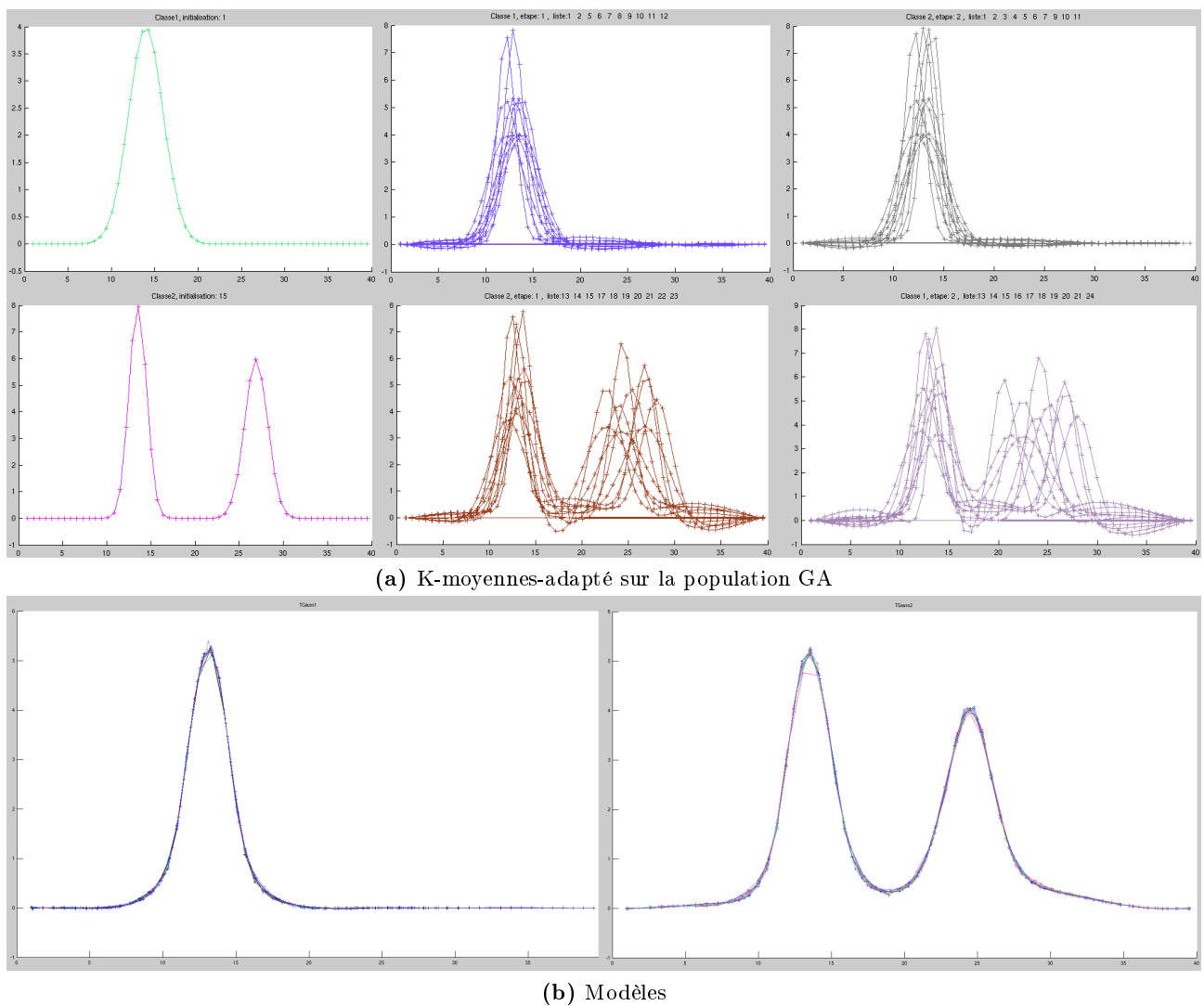


FIG. 30: (a) Dès la première itération les courbes sont bien classées, pratiquement toutes les courbes de la population apparaissent dans les classes, ce qui est normal puisque les classes de la population A sont assez bien séparées. L'algorithme s'arrête au bout de 2 itérations, car il n'y a pas de changements entre les deux classes. (b) Les deux modèles calculés à partir des deux ensembles de courbes trouvés en (a), avec la méthode d'estimation de modèle vue dans la partie 3. Ce sont effectivement les deux types de courbes présents dans la population GA.

tion est choisie au hasard dans le domaine de définition des points, donc les centres ne sont pas des points, mais entre les points, d'où l'idée de prendre deux courbes et de prendre son modèle comme premier centre, pour chaque cluster. En réalité ça n'améliore pas l'initialisation, puisque pour que le résultat soit vraiment meilleur, il faudrait que le modèle du centre initial soit composé de deux courbes de la même classe.

5.6 Discussion

L'utilisation du K-moyennes dans sa forme initiale posait des problèmes liés aux distances et à l'instabilité face à l'initialisation. Pour résoudre ces problèmes, nous avons proposé des changements dans l'algorithme afin que les distances soient mieux traitées, et que l'initialisation puisse être corrigée automatiquement. Cela a toutefois nécessité l'introduction d'autres paramètres, en plus du paramètre K . Les paramètres *marge* et *seuil* sont à définir en fonction de la taille des données. Pour les populations GA et GB le *seuil* choisi était 1/3 de la population, et la *marge* était de 2 (suffisant pour séparer les données de GB). Si K est connu par avance, il serait juste de faire dépendre le paramètre *seuil* de la taille des données, mais aussi de K , car si $K = 10$, il n'est pas conseillé de prendre un seuil valant le tiers de la taille des données. Je pense qu'il est raisonnable de prendre un *seuil* de l'ordre de $\frac{\text{taillePopulation}}{K}$. Pour la *marge*, tout dépend des données elles-mêmes : sont-elles bien séparées ? Il n'est a priori pas facile ni possible de répondre à cette question. Il est donc plutôt raisonnable de ne pas prendre une marge trop petite. Pour séparer 36 sujets, une marge de 2 était suffisante, une marge de 3 fonctionne aussi bien, mais classe moins de sujets. Au contraire, une marge de 1 mettra plus de temps à converger si l'initialisation est mauvaise et si il y a de mauvais classements, ce qui avec une si petite marge peut arriver. Bien évidemment la marge doit être plus petite que le seuil si on veut espérer avoir au moins un sujet par classe. Sans a priori sur les données, prendre une marge représentant un dixième du seuil paraît être raisonnable.

Le paramètre définissant le nombre de classes K peut être plus délicat à fixer. Dans notre cas, ce paramètre n'est pas du tout connu, c'est d'ailleurs un des problèmes principaux. Il existe des méthodes d'estimation du nombre de K comme les articles de Kothari et al. en 1999 [19] et Lu et al. en 2008 [21]. D'autres comme Pelleg et al. en 2000 [27] proposent (méthode du x-means) de donner un K_{max} en paramètre, qui représente le nombre de classes maximum que peut trouver l'algorithme. La méthode commence par séparer les données en deux avec un K-moyennes standard, puis teste si l'une des deux parties est encore "séparable", jusqu'à ce que soit les données ne soient plus "séparables", soit que le nombre de classes atteigne K_{max} . Par manque de temps je n'ai pas pu adapter cette méthode de la même façon qu'à été adaptée la méthode des K-moyennes, mais il serait intéressant de voir ce qu'elle donne. On risque toutefois de se heurter à un problème de temps de calcul, la méthode adaptée du K-moyennes étant déjà assez coûteuse en temps.

6 Analyse en Composantes Principales des vecteurs moments initiaux

L'approche que nous allons présenter maintenant part d'un modèle unique calculé à partir de l'ensemble de la population, contrairement à la précédente qui calculait itérativement des modèles. Cette partie correspond à la branche centrale du graphique à la figure 4 de la page 10.

6.1 Les vecteurs moments initiaux

Avant toutes choses, il faut donc calculer le modèle commun à la population avec la méthode vue en 3.4 page 21. Il est évident que si la population comporte plusieurs modes, comme par exemple la population GB qui comporte un mode avec des gaussiennes à une bosse, et un autre mode avec des mélanges de deux gaussiennes, le modèle commun à toute la population (figure 32) ne sera représentatif d'aucun individu de la population, car il ne prend pas en compte l'existence de différents modes. Par contre, ce sont les déformations conduisant à ce modèle qui vont nous intéresser. De la création de ce modèle unique, nous pouvons récupérer les vecteurs moments de tous les individus le long de la déformation. A chaque instant t de la déformation, les vecteurs moments de chaque individu représentent les tangentes des trajectoires.

On a vu dans la section 3 qu'on avait la relation suivante entre une déformation et son inverse. Soient $x \in \mathbb{R}^d$, v un champ de vecteurs, et la déformation ϕ :

$$\phi_t^{-1} \circ \phi_t(x) \xrightarrow[t \rightarrow 1]{} x, \text{ avec rappelons : } \phi_t^v(x) = x + \int_0^t v_s \circ \phi_s^v(x) ds \quad (22)$$

Donc quand $t = 1$, c'est à dire quand la déformation est finie, appliquer l'inverse de cette déformation au modèle déformé nous ramène au modèle original. Les vecteurs vitesses finaux de la transformation ϕ sont donc les vecteurs vitesses initiaux de la transformation inverse ϕ^{-1} .

Rappelons aussi la relation qui lie vecteurs vitesses v et vecteurs moments α . Avec $x_i(t)$ l'instant t de la trajectoire du sujet $i \in \{0; N\}$:

$$\frac{dx_i}{dt} = v_t(x_i) = K_V(x_i, \cdot) \alpha_i \iff v_t = K_V * \alpha \quad (23)$$

$$\iff \alpha = K_V^{-1} * v_t \quad (24)$$

$$\implies \gamma_i = \sum_{i=1}^N K_V(x_i, \cdot) \alpha_i$$

Les vecteurs moments finaux d'une déformation, sont donc les opposés des vecteurs moments initiaux de l'inverse de cette même déformation.

Les vecteurs vitesses initiaux sont important car déterminent la trajectoire de la déformation. Intuitivement, lors d'un lancer de pierre, la trajectoire que va effectuer la pierre est déterminée par la vitesse initiale mise dans le lancer. Lors du recalage d'une surface S sur un modèle T , on cherche à minimiser la fonctionnelle J suivante :

$$J(v_t) = \int_0^1 \|v_t\|^2 dt + A(\phi \circ S, T)$$

où A est le terme d'appariement, qui cherche le difféomorphisme optimal tel que S soit amené le plus proche possible de

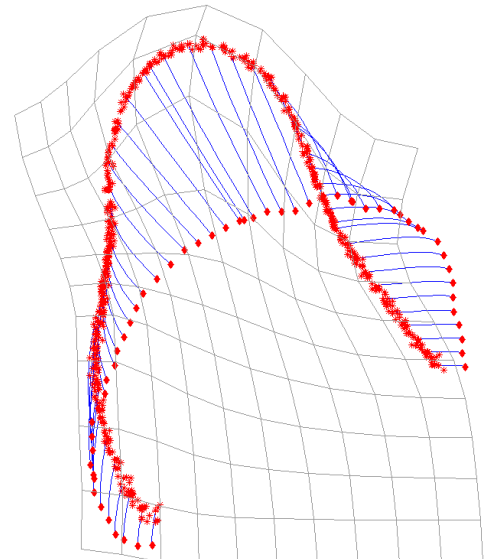


FIG. 31: trajectoires (en bleu) des points rouges au modèle '*'

T. Si les positions finales des points $\phi \circ S$ sont fixées, le terme d'appariement devient une constante, on veut alors minimiser l'intégrale de la fonctionnelle avec $\phi \circ S$ fixé. $x(t)$ représentent les points de la surface à l'instant t .

$$\begin{aligned} \int_0^1 \|v_t\|_V^2 dt &= \int_0^1 \frac{dx(t)}{dt}^T K_V^{-1}(x(t), x(t)) \frac{dx(t)}{dt} dt \\ &= \int_0^1 \underbrace{\left\| \frac{dx(t)}{dt} \right\|_{K_V^{-1}}^2}_{\text{norme pour la métrique } K_V^{-1}} dt \end{aligned} \quad (25)$$

On veut donc minimiser $\int_0^1 \left\| \frac{dx(t)}{dt} \right\|_{K_V^{-1}}^2 dt$ en connaissant $x(0) = S$ et $x(1) = \phi \circ S$, ce qui revient à trouver le plus court chemin pour la métrique K^{-1} , qui lui vérifie une équation géodésique f :

$$\begin{cases} x''(t) = f(x, x', t) \\ x(0), x(1) \text{ connus} \end{cases}$$

Une fois ce problème résolu, on peut conserver les valeurs $x'(0)$. Les $x'(0)$ sont les dérivées des points de la surface par rapport au temps en $t = 0$, c'est-à-dire exactement les vecteurs vitesses initiaux aux points $x(0)$. On pourra ensuite recalculer $x(t)$ en résolvant :

$$\begin{cases} x''(t) = f(x, x', t) \\ x(0), x'(0) \text{ connus} \end{cases}$$

C'est ce problème qui est utilisé dans la méthode de tir géodésique fournie par Joan Glaunès.

Nous appelons maintenant les vecteurs moments initiaux l'opposé des vecteurs moments finaux des déformations de chaque individu constituant le modèle.

Les vecteurs moments ont un autre intérêt : le calcul de la norme des vecteurs vitesses nécessite d'inverser le noyau alors que l'utilisation de vecteurs moments ne nécessite pas cette inversion. De plus, les vecteurs moments ne sont définis que sur les points des surfaces alors que les vecteurs vitesses sont définis sur tout l'espace, car c'est tout l'espace qui se déforme.

6.2 L'Analyse en Composantes Principales à noyau

On cherche à analyser les vecteurs moments initiaux de ce modèle unique, afin d'observer la répartition des formes autour du modèle : sont-elles uniformément distribuées autour ou non ? Pour cela nous avons besoin de modifier un peu la méthode standard d'ACP dont voici le principe :

- centrer les données sur leur centre de gravité en soustrayant à chaque variable sa moyenne.
- calculer ensuite la matrice de variance-covariance des données centrées.
- trouver les axes qui maximisent la variance des projections des données sur ces axes. Le premier axe est le vecteur propre dont la valeur propre (qui correspond à la variance empirique du vecteur propre associé) est la plus grande. Il faut donc calculer les valeurs propres et les vecteurs propres de la matrice de variance-covariance.

Dans notre cas nous avons des vecteurs moments définis sur tous les sujets en chaque point de la population : c'est-à-dire que par exemple, pour une population de 10 sujets comprenant chacun 100 points définis en 3D, l'ensemble des vecteurs moments pour chaque sujet sont de taille $3 \times (1000)$. Ces coordonnées sont mises bout à bout et on obtient la matrice suivante :

$$A = \begin{pmatrix} x_1^1 & y_1^1 & z_1^1 & x_1^2 & y_1^2 & z_1^2 & \dots \\ x_2^1 & y_2^1 & z_2^1 & x_2^2 & y_2^2 & z_2^2 & \dots \\ x_3^1 & y_3^1 & z_3^1 & x_3^2 & y_3^2 & z_3^2 & \dots \\ x_4^1 & y_4^1 & z_4^1 & x_4^2 & y_4^2 & z_4^2 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

où x_i^j représente la première coordonnée du point j du sujet i , y_i^j la seconde, et z_i^j la troisième. La matrice des données A a donc une taille de $nbSujet \times (nbDim * nbPoints)$, avec $nbPoints$ le nombre total de points dans la population.

De plus nos vecteurs moments ne sont pas dans le même espace que les vecteurs vitesses, ils sont dans l'espace des déformations. Donc pour le calcul de la matrice de covariance, on a besoin du noyau de taille $nbSujet \times nbSujet$ qui a permis de les construire. Pour calculer cette matrice, il faut adapter le noyau, car les données ne sont plus représentées sur $nbDim$ lignes, mais sur une seule. Chaque coefficient K_{ij} de K est remplacé par un bloc de taille $nbDim \times nbDim$ avec les coefficients correspondant sur la diagonale et des zéros partout ailleurs : $MatCov = A * K_V * A^T$.

La suite de l'algorithme est le même que pour l'ACP standard.

Rappelons qu'un hippocampe contient environ 3000 points, l'une des populations d'hippocampes (S1S2A) contient 10 sujets, et l'autre en a 30 (S1S2B). Ce qui fait des populations comprenant environ 30 000 points pour S1S2A et 90 000 points pour S1S2B. On peut facilement se rendre compte que de manipuler des matrices de $nbDim * 30000 \times nbDim * 30000$ (pour les noyaux) et de $10 \times (3 * 30000)$ (pour la matrice des données) peut poser des problèmes de mémoire. Afin d'alléger les données, nous avons décidé d'utiliser un Matching Pursuit sur le modèle de la population (cf figure 32). Le noyau

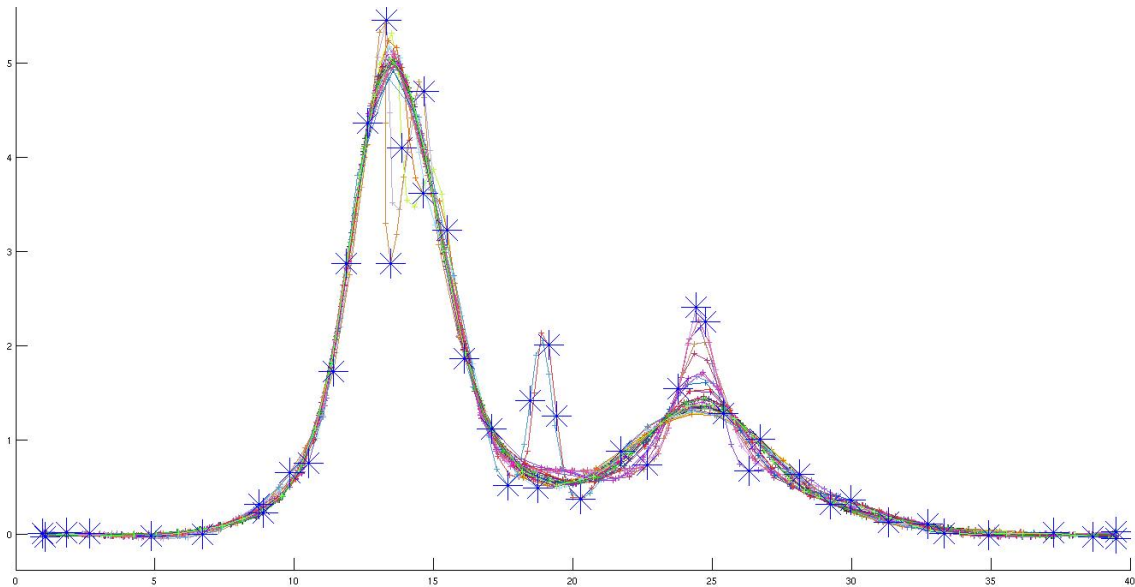


FIG. 32: *Modèle unique de la population B des gaussiennes. Les étoiles bleues sont les points gardés par le modèle pour une précision de 10^{-11} . Le modèle passe de 2016 points à 42. On peut remarquer au passage que le modèle n'est effectivement pas représentatif de la population.*

est ensuite recalculé à partir des points conservés, il est maintenant plus petit, ainsi que les vecteurs moments.

Le problème est que pour les populations comportant beaucoup de points, comme la population S1S2B (cf page 25), il n'est même pas possible de calculer un noyau de taille 90000×90000 pour le Matching Pursuit car pose des problèmes de mémoire. Pour cela Joan Glaunès a fourni des codes

Cuda pour un calcul de noyau optimisé sur GPU.

Pour mieux se rendre compte de la signification de l'axe principal, nous avons effectué des tirs géodésiques à partir du modèle de la population dans la direction (positive ou négative) de l'axe principal. Autrement dit, les vecteurs initiaux aux points du modèle pour le tir géodésique sont des combinaisons linéaires des vecteurs moments initiaux utilisés dans l'ACP avec les coefficients du premier vecteur propre.

6.3 Résultats

Les résultats sont présentés sur les figures allant de 33 à 38. La figure 33 correspond au résultat de l'ACP sur la population GB : on voit qu'il y a un recouvrement entre les deux classes qui ne sont en effet pas bien séparées. En revanche, l'ACP permet tout de même de bien capturer la variabilité au sein de la population comme on peut le voir lorsqu'on fait un tir géodésique le long de l'axe principal (figure 34). Sur les différentes populations d'hippocampes, on constate que les différents sujets sont très bien séparés dans l'espace de l'ACP (figures 35 et 36). Les figures 37 et 38 présentent les résultats dans le cas de la population GC qui a 3 classes. Le premier résultat de l'ACP montre des classes assez peu séparées. Lorsqu'on fait une nouvelle ACP sur les sous-populations correspondant aux valeurs positives et négatives de l'axe principal la séparation est un peu meilleure.

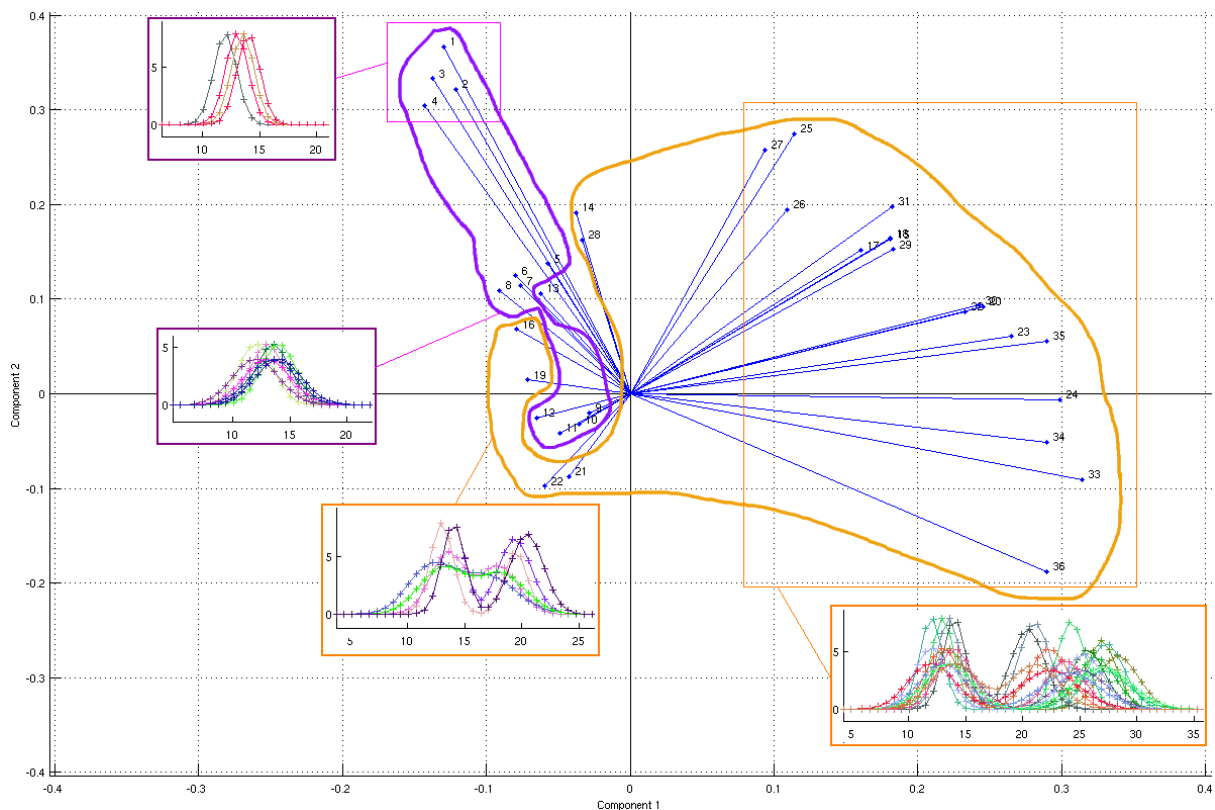


FIG. 33: ACP à noyau sur la population GB. Les points entourés de violet sont les courbes à une bosse, ceux qui sont entouré en orange correspondent aux courbes à deux bosses. Dans les encadrés on peut voir les courbes correspondantes. On remarque que les "une bosse" ne sont pas très bien séparés des "deux bosses". Cela vient du fait que la population GB est mixte et pas bien séparée, elle contient des courbes (les numéros 16 19 21 et 22) ayant presque deux bosses. Quant aux courbes 13 14 et 28, leurs deux bosses sont proches et leurs amplitudes importantes. On remarque aussi que le long de l'axe 2, les courbes dont l'amplitude est plus importante se trouvent en haut, et celles dont l'amplitude est plus faible sont en bas.

6.4 Discussion

Bien que certaines populations soient très mal séparées du fait d'un continuum entre les classes et que la seule visualisation du résultat de l'ACP ne permet pas de dire combien il y a de classes. Mais en faisant un tir géodésique dans la direction de l'axe principal, on peut déjà extraire deux tendances de la population si la population présente plusieurs classes, les tirs géodésiques permettent de se rendre compte visuellement de ce qu'explique la première composante de l'ACP à noyau. S'il y a à droite une tendance à avoir deux bosses et à gauche une bosse comme montré à la figure 34, on peut dire qu'il y a au moins deux classes. En effet, un tir géodésique sur l'axe principal de la figure 37 ne donne pas tout à fait dans le sens négatif deux bosses et dans le sens positif une bosse sur la droite (de l'axe des abscisses des courbes), car la répartition n'est pas exactement le long de l'axe principal, mais les tirs géodésiques donnent une nouvelle forme vers laquelle tous le modèle de la population se déplace ; s'il n'y avait pas autant de variabilité dans la population, ce qui arrive quand la population ne se compose que d'une classe : les tirs géodésiques ne donnent pas l'impression d'aller dans le même sens, en chaque point du modèle le sens de la déformation est différent, les déformations ne sont plus globales mais locales (du à la variabilité intra classe), les changements de formes expliqués par l'axe principal ne sont pas globaux (comme dans le cas de plusieurs classes de formes), mais locaux. Nous savons alors qu'il y a au moins deux classes. En regardant les sujets ayant une coordonnée négative sur l'axe principal, nous arrivons à la figure 33, qui nous donne dans le sens négatif une bosse sur la gauche et dans le sens positif, deux bosses. En faisant de même sur les données ayant une coordonnée positive sur l'axe principal de la figure 37, on obtient une répartition du type de la figure ??, avec un tir dans le sens négatif de l'axe principal donnant une bosse sur la droite de l'axe des abscisses, et un tir dans le sens positif donne une bosse encore plus à droite. La population de la figure 37 a donc bien 3 classes, qui sont dans l'ordre des tirs : une bosse à gauche, deux bosses, une bosse à droite.

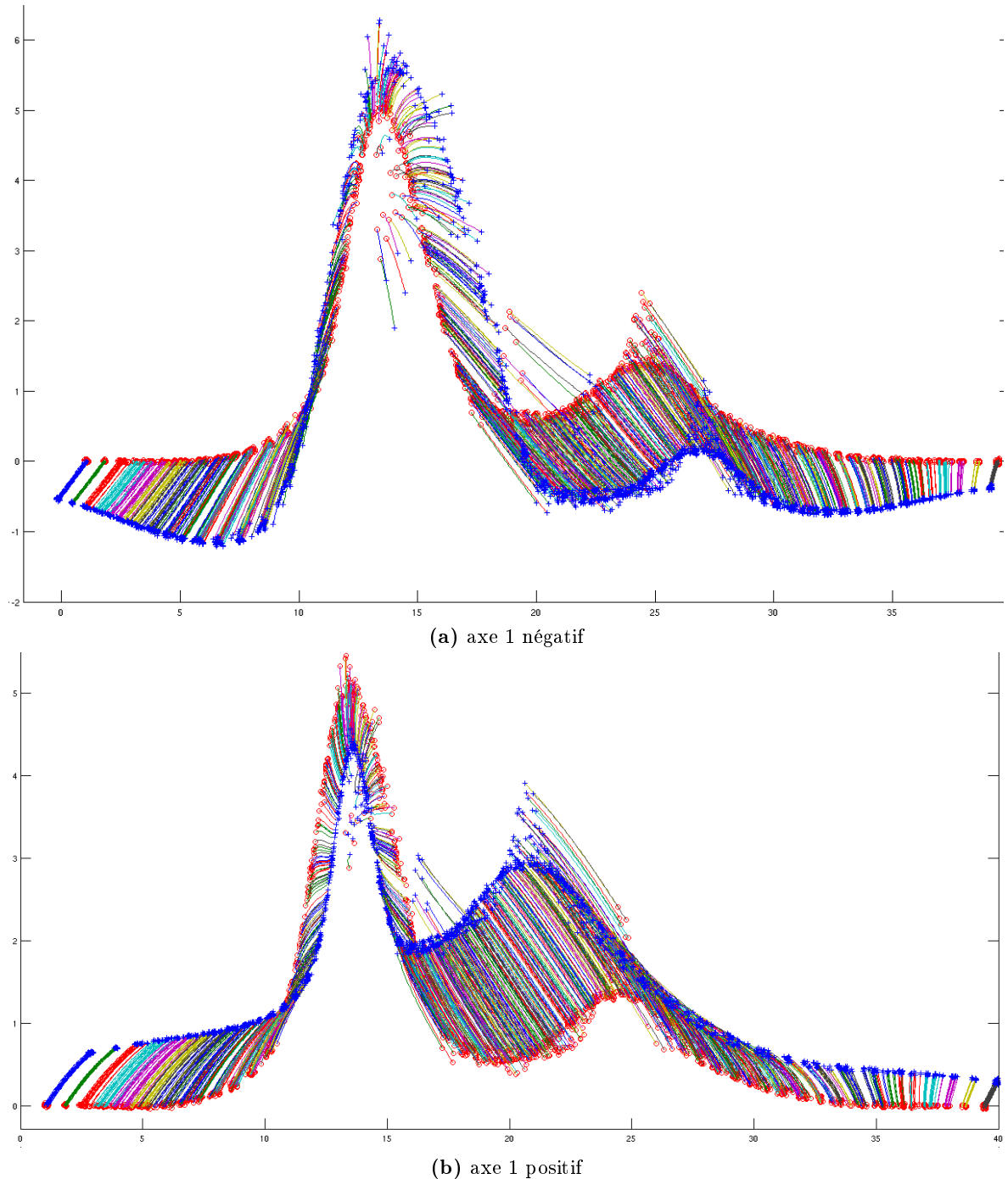


FIG. 34: Tirs géodésiques le long de l'axe 1 de la figure 33 depuis le modèle commun de la population de courbes GB, qui apparaît en rouge. En bleu, point d'arrivée du modèle après un tir le long de l'axe principal. (a) Dans le sens négatif de l'axe principal, le modèle perd sa seconde bosse. (b) Dans le sens positif, la seconde bosse prend de l'importance tandis que la première diminue. Ainsi, même si les classes ne sont pas séparables, on voit que l'axe principal capture bien la variabilité de la population.

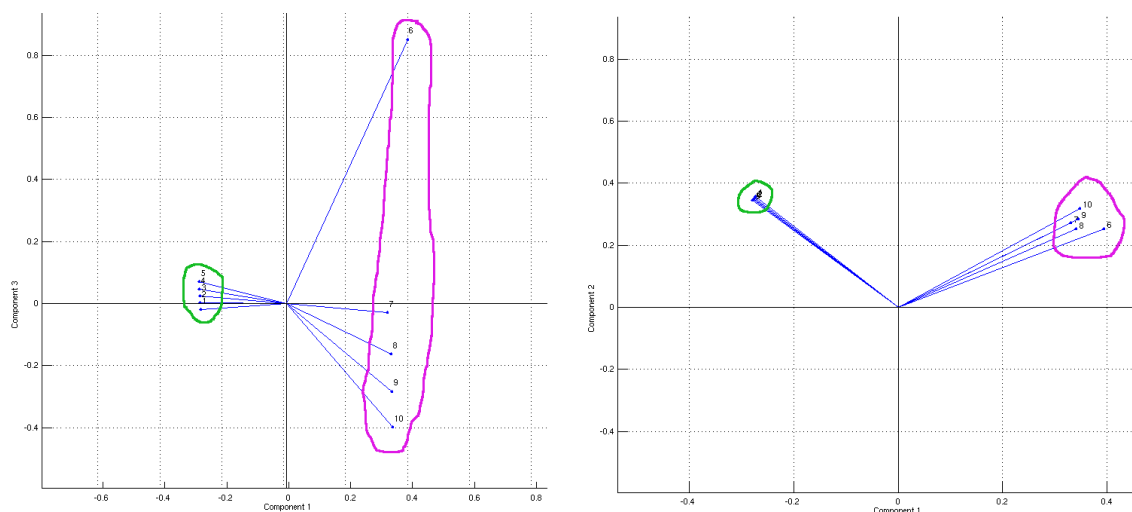
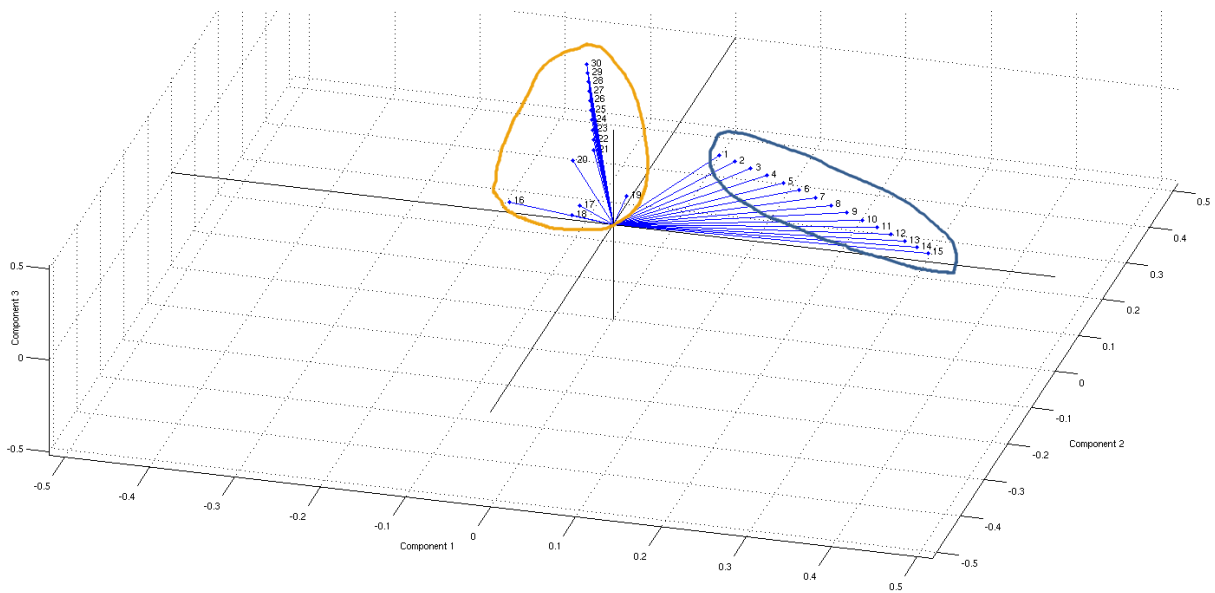
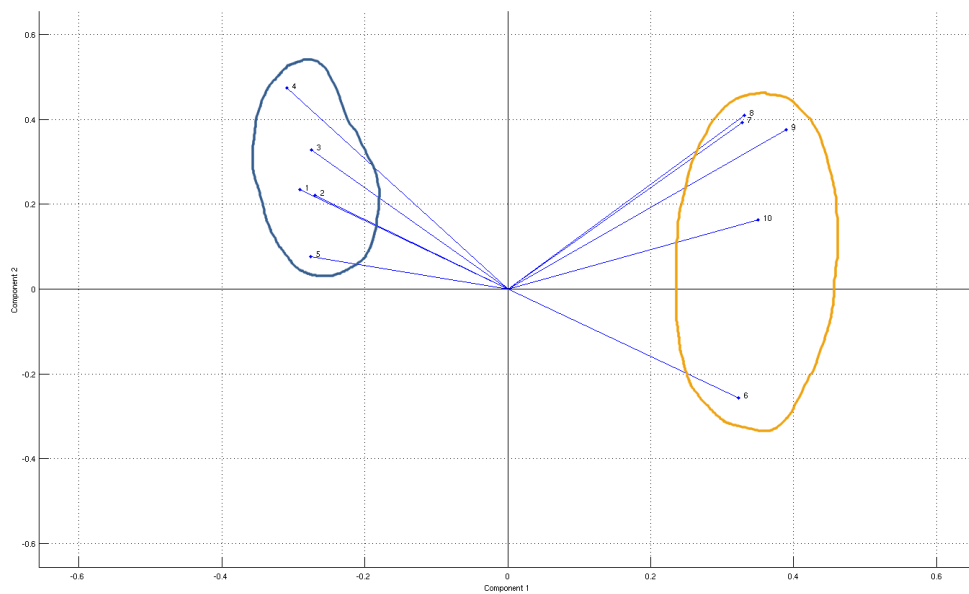


FIG. 35: ACP à noyau sur la population S1S2A. A gauche, l'axe des abscisses est la première composante principale, et l'axe des ordonnées est la troisième. A droite, il s'agit de la première en abscisse et de la deuxième composante principale en ordonnée. En vert sont entourés les points correspondant aux 5 hippocampes de la classe S1A et en rose les 5 correspondant aux hippocampes de la classe S2A. Les points sont très bien séparés, ce qui est plutôt normal puisque la population S1S2A est faite à partir de deux hippocampes très peu déformés. De plus on peut remarquer le continuum du à la création de la population S1S2A.



(a) résultat ACP à noyaux sur S1S2B



(b) résultat ACP à noyaux sur S1S2C

FIG. 36: En (a), on voit clairement la caractéristique de la population S1S2B qui est formée de deux continnuums issus des deux hippocampes S1 et S2. Les couleurs entourent les sujets S1 en bleu, et S2 en jaune.

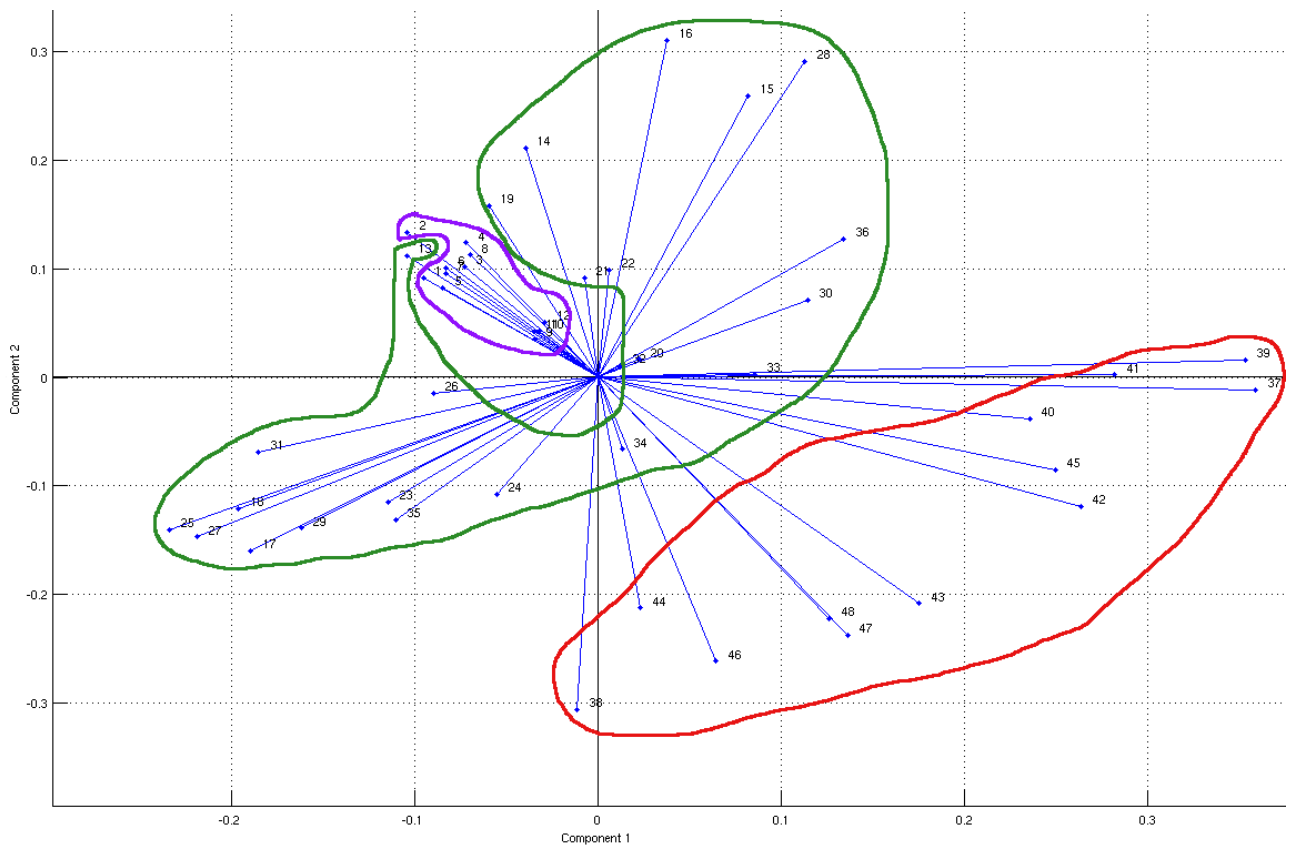
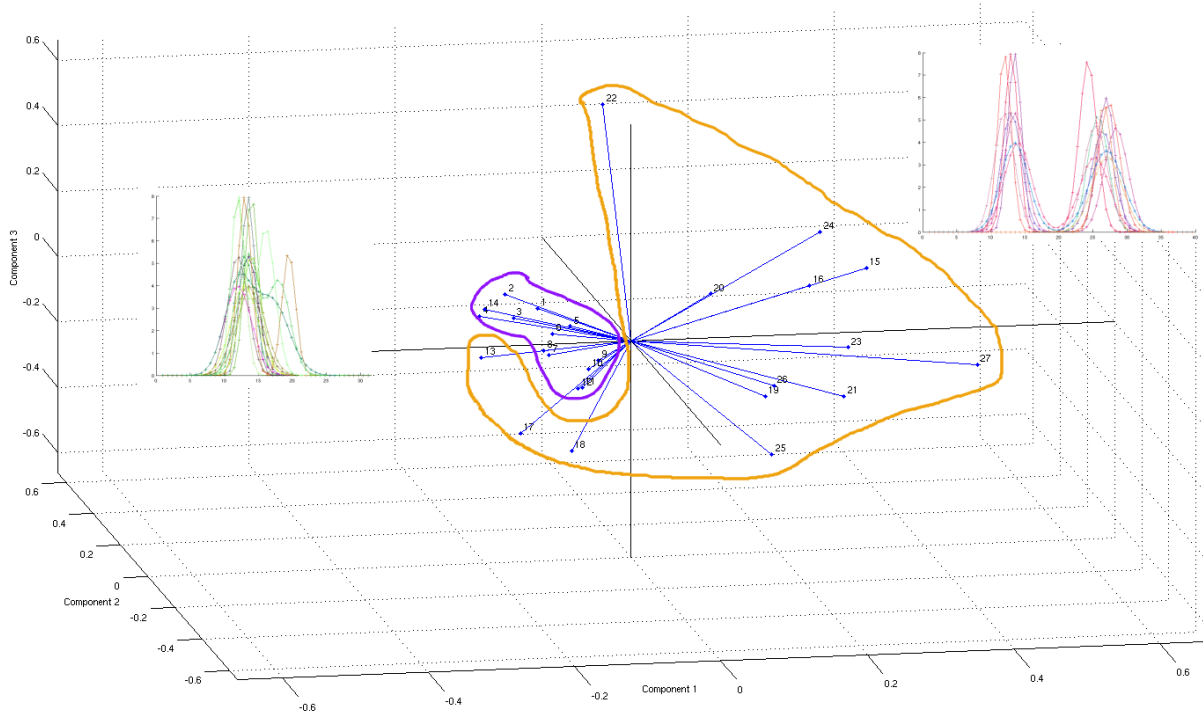
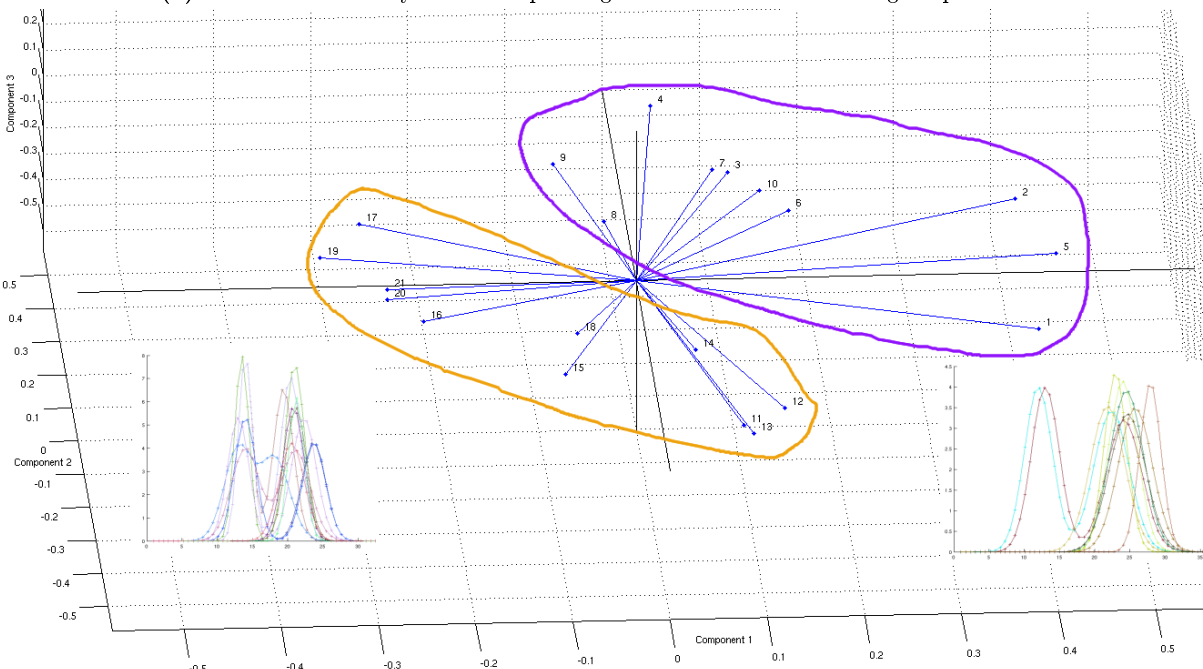


FIG. 37: Population GC à 3 classes : en violet sont entourées les « une bosse à gauche », en vert les « deux bosses », et en rouge les « une bosse à droite ». les résultats à la figure suivante (figure 38) sont issus de l'analyse faite sur les populations comptenant les sujets à gauche de l'axe vertical, et l'autre les sujets se trouvant à droite.



(a) résultat ACP à noyaux sur la partie gauche des courbes de la figure précédente



(b) résultat ACP à noyaux sur la partie droite des courbes de la figure précédente

FIG. 38: En (a), l'ACP à noyau a été effectuée sur les courbes ayant le coefficient de la première composante négatif, c'est-à-dire des courbes avec un ou deux bosses (entourées de jaune). On observe la répartition à droite de courbes avec deux bosses, le graphique en haut à droite représente toutes les courbes se trouvant à droite du second axe; à gauche nous avons les courbes à une bosse ainsi que les deux bosses rapprochées, la séparation n'est pas claire, mais un tir géodésique le long de l'axe principal, donne le même genre de résultats qu'à la figure 34, ce qui permet de constater deux tendances. En (b), l'ACP à noyau a été effectuée sur les courbes ayant le coefficient de la première composante positif, c'est-à-dire des courbes avec deux (entourées en jaune) ou trois bosses. On remarque que la séparation est plus nette, mais ne se fait pas avec uniquement le premier axe.

7 Méthodes utilisant une matrice de distances

Les approches de cette partie reposent sur des recalages deux-à-deux entre les individus de notre population (branche de gauche de la figure 4). Ces recalages permettent d'obtenir une matrice de similarité entre les différents individus sur laquelle sont appliqués des algorithmes de réduction de dimension pour pouvoir classer ensuite les individus dans le nouvel espace.

7.1 Estimation de la matrice de distance

Dans notre cas, une matrice de distance de la population est en fait la matrice regroupant toutes les distances de déformation d'un individu à un autre. Donc pour une population de n individus, au moins $n/2$ recalages sont nécessaires, si l'on calcule seulement la moitié de la matrice (l'autre moitié est déduite par symétrie). Voici une idée des temps de calcul sur CPU et GPU d'un recalage, du calcul d'une matrice de distance 10×10 (donc 50 recalages) et de 100×100 (donc 5000 recalages)

TAB. 1: Temps de calculs sur CPU et GPU pour 1 recalage, pour une matrice de distances 10×10 puis 100×100

Recalage(s) du sujet S1 au sujet S2	Calcul sur CPU	Calcul sur GPU
1 recalage	4200 secondes	45 secondes
50 recalages	3500 min \approx 58h20	37 min 30 s
5000 recalages	5833.3 h \approx 243 jours \approx 8 mois	62.5 h \approx 2 jours 14h

Il est donc intéressant d'essayer d'approximer le calcul de matrice de distances, comme l'ont proposé Yang et al. [33]. Ils ont montré que les approximations de premier et second ordre de la métrique difféomorphique étaient proches de la métrique complète, et que ces approximations augmentent considérablement la rapidité de calcul de la matrice de distances. Leur test sur une base de 196 patients nécessite 6 jours pour la métrique entière contre 1 heure pour les approximations de premier et second ordre.

Le principe des approximations proposées dans [33] est le suivant : à partir d'un modèle unique, donc à partir des déformations des sujets au modèle, soit $nbSujets$ déformations, approximer les déformations entre les sujets deux à deux. Nous décrivons brièvement la méthode ci-dessous.

Les approximations des mesures difféomorphiques s'appuient sur la théorie de Lie et sur le fait qu'un difféomorphisme ϕ_t est un flot d'équations différentielles ordinaires pour lequel $\frac{d\phi_t}{dt} = v_t(\phi_t)$ et $\phi_0 = Id$. Il en suit que la transformation ϕ_t^v est l'application exponentielle du vecteur vitesse au temps t notée :

$$\phi_t^{v_0} = Exp_{Id}(tv_0)$$

Soient $\phi_t^{v_0^i}$ la transformation difféomorphique qui emmène une forme T sur une forme S_i et $\phi_t^{v_0^j}$ la transformation difféomorphique qui emmène une forme T sur une forme S_j . On a par composition la transformation de S_i à S_j : $S_j = (\phi_1^{v_0^j} \circ (\phi_1^{v_0^i})^{-1}) \circ S_i$. Donc à $t = 1$, la géodésique reliant S_i et S_j est donnée par :

$$\phi_1 = Exp_{Id}(v_0^j) \circ Exp_{Id}(-v_0^i)$$

En posant $a = Log_{Id}(Exp_{Id}(v_0^j) \circ Exp_{Id}(-v_0^i))$ et en utilisant la formule de Baker-Campbell-Hausdorff

qui résout une équation du type $\log(\exp^x \exp^y)$ avec x et y des éléments d'une algèbre de Lie, on a :

$$a = v_0^j - v_0^i + 1/2[v_0^j, -v_0^i] + \dots,$$

avec $[\cdot, \cdot]$ le crochet de Lie.

On a bien $\phi_1 = \text{Exp}_{Id}(a) = \text{Exp}_{Id}(v_0^j - v_0^i + 1/2[v_0^j, -v_0^i] + \dots)$.

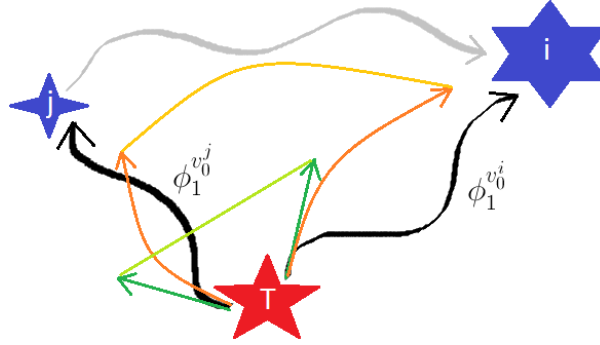


FIG. 39: Illustration montrant en vert pâle l'approximation au premier ordre de la distance entre les individus i et j de la population, en jaune l'approximation au second ordre et en gris la vraie distance entre les sujets. L'étoile rouge est le modèle commun à la population. Les traits noirs sont les déformations allant du modèle aux individus i et j de la population. Les traits verts sont les approximations au premier ordre des déformations noires, et les oranges sont leurs approximations du second ordre.

L'approximation du premier ordre est donc :

$$\phi_1 = \text{Exp}_{Id}(a) \approx \text{Exp}_{Id}(v_0^j - v_0^i) = \phi_1^{v_0^j - v_0^i} \quad (26)$$

L'approximation de la transformation du sujet S_i au sujet S_j est donnée par une transformation dont le vecteur vitesse initial est $v_0^j - v_0^i$. Ce qui est effectivement une importante approximation. La distance de cette déformation est aussi approximée et vaut :

$$\left\langle (v_0^j - v_0^i), (v_0^j - v_0^i) \right\rangle_V$$

L'approximation du second ordre est donnée par :

$$\phi_1 = \text{Exp}_{Id}(a) \approx \text{Exp}_{Id}(v_0^j - v_0^i + 1/2[v_0^j, -v_0^i]) \quad (27)$$

Ce qui donne l'approximation de la distance suivante :

$$\left\langle (v_0^j - v_0^i + 1/2[v_0^j, -v_0^i]), (v_0^j - v_0^i + 1/2[v_0^j, -v_0^i]) \right\rangle_V$$

Au vu du peu de différences qu'il y a entre l'approximation du premier et du second ordre (table 2), nous avons choisi l'approximation du premier ordre qui est beaucoup plus simple à mettre en place, et moins coûteuse.

Nous l'avons testé sur la population de courbes GA. La matrice de distances approximée au premier ordre semble une bonne approximation de la vraie matrice de distances : les distances ne sont pas du même ordre de grandeur, mais leur ordonnancement reste quasiment inchangé à quelques exceptions près qui sont en fait des inversions. Pour 36 sujets, les 12 premiers sujets les plus proches du sujet 1 sont exactement dans le même ordre que ce soit avec la matrice approximée à l'ordre 1 ou la vraie matrice de distances calculée à partir de recalages deux à deux. La suite du classement

TAB. 2: Table issue de [33] présentant la méthode d'approximation en question. Les matrices des distances totales, des distances approximées au premier ordre et au second ordre ont été utilisées dans la méthode ISOMAP (voir page 58). La première ligne correspond à la corrélation entre la première dimension de l'ISOMAP et l'âge des sujets, la seconde ligne est la corrélation avec cette fois-ci la seconde dimension.

ISOMAP embedding	Full metric	1 st order $\rho^{1^{st}}$	2 nd order $\rho^{2^{nd}}$
1st dimension	0.65	0.6	0.59
2nd dimension	0.1	0.04	0.05

est à peu près le même, quelques sujets sont bien classés, les autres sont simplement des inversions. Ceci n'est pas très important, car ce qui nous intéresse sont en effet les sujets les plus proches pour chacun d'entre eux et non les plus éloignés.

Le temps de calcul de la matrice de distances des courbes de la population GB (36 courbes de 56 points), qui sont donc de très petites données, est de 25 minutes ; et passe à seulement 30 secondes pour le calcul des approximations au premier ordre des distances. Le temps de calcul de la matrice de distances approximées de la population S1S2C qui contient 10 sujets, est d'environ 8 minutes sur GPU. Le temps de calcul de l'approximation par rapport au temps de calcul des distances complètes décroît avec le nombre de sujets.

7.2 Spectral Clustering

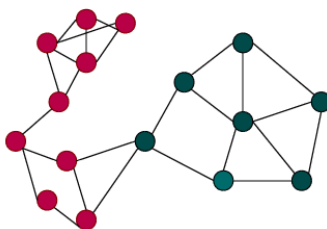


FIG. 40: Le graphe représente un 3-NN sur une population de points comportant deux classes : points rouges et points verts. Seulement 3 voisins, et tous les individus de la population sont reliés.

Maintenant que nous savons calculer rapidement une matrice de distances, nous pouvons appliquer différentes méthodes d'analyse. Notre idée initiale était d'utiliser la méthode des K plus proches voisins (appelée K -NN pour K -Nearest Neighbors). Le K -NN fonctionne plutôt bien si l'on choisit bien le nombre K de voisins et si les données sont suffisamment bien séparées. Ces conditions ne sont bien évidemment pas remplies dans notre cas, surtout la condition touchant à la séparation des données. Des classes peu ou mal séparées, avec un nombre de voisin un peu trop grand, conduisant à regrouper toutes les classes (Figure 40).

Nous nous sommes donc plutôt tournés vers le Spectral Clustering, qui résout ce problème en coupant le graphe.

7.2.1 Méthode

Le Spectral Clustering, méthode présentée par Ng et al en 2001 [26] contourne ce problème de classes trop proches ou mal séparées. La méthode consiste à partitionner le graphe des données en un nombre de classes tel que la similarité intra-classes est haute et la similarité inter-classes est basse. L'idée du Spectral Clustering est de trouver la coupe minimale d'un graphe tel que celui présenté à la figure 40. Voyons de plus près cette méthode.

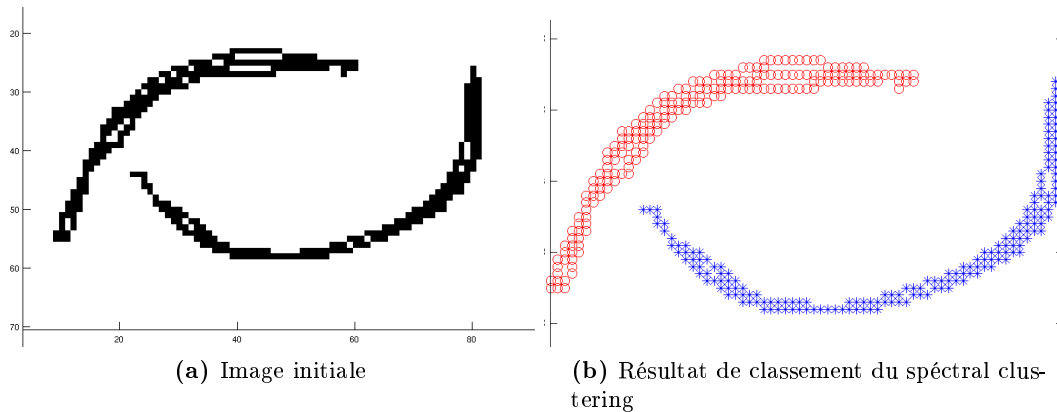


FIG. 41: (a), Image initiale. (b), Résultat du Spectral Clustering sur l'image. Les étoiles bleues représentent les points classés dans la première classe, les ronds rouges ceux classés dans la seconde.

Le premier pas consiste à construire le graphe. Pour cela, il faut d'abord calculer la matrice de similarité, qui est elle calculée à partir de la matrice des distances $Dist$. Dans notre cas, nous avons choisi de calculer la matrice de distances approximées à l'ordre 1 et comme mesure de similarité nous avons choisis un noyau gaussien : $Sim(i, j) = e^{-\frac{Dist(i, j)}{2\sigma^2}}$ pour le calcul de la matrice de similarité Sim , afin de donner plus d'importance aux éléments proches.

Il faut ensuite construire la matrice d'adjacence A du graphe représentant les données. Pour ce faire, on peut soit choisir un seuil en dessous duquel la similarité sera considérée comme trop faible, ainsi il n'y aura pas d'arc entre deux sommets ayant une similarité trop peu significative ; soit en choisissant un nombre de voisins (le K requis pour l'algorithme du K-NN), façon pour laquelle nous avons opté arbitrairement. Ce nombre ne doit pas être trop petit pour que toutes les composantes du graphe soient connectées, nous avons pris un voisinage de 5 pour 36 individus. On sait que dans une matrice d'adjacence si $A(i, j) = 0$ il n'y a pas d'arc dans le graphe qui relie les sommets i et j . Si $A(i, j) = Sim(i, j)$, l'arc liant les sommets représentant les individus i et j vaut la similarité entre i et j .

Maintenant que le graphe est construit, calculons sa matrice laplacienne L :

$$L = D.A.D, \quad (28)$$

avec D la matrice diagonale des degrés de chaque sommet.

Il ne reste plus qu'à calculer les valeurs propres avec les vecteurs propres associés de la matrice L . Les n vecteurs propres (avec n représentant le nombre de classes qu'on veut faire apparaître) dont les valeurs propres sont les plus grandes sont stockés dans une matrice V de dimension $nbSujet \times n$. La méthode se termine par l'utilisation d'un algorithme K-moyennes sur la matrice V le long de ses colonnes.

La figure 41 montre le résultat sur une image de test en 2 dimensions, afin d'illustrer la méthode.

7.2.2 Résultats

Voici quelques résultats de Spectral Clustering sur une population de courbes à deux classes, une à trois classes, et une dernière d'hippocampes à deux classes.

La population de courbe GB (Figure 42) est présentée dans l'espace V puis dans sa version normalisée. Les résultats d'après sont tous présentés dans l'espace normalisé : le résultat sur la population formée de trois classes utilisées dans l'ACP à noyau est en figure 43.

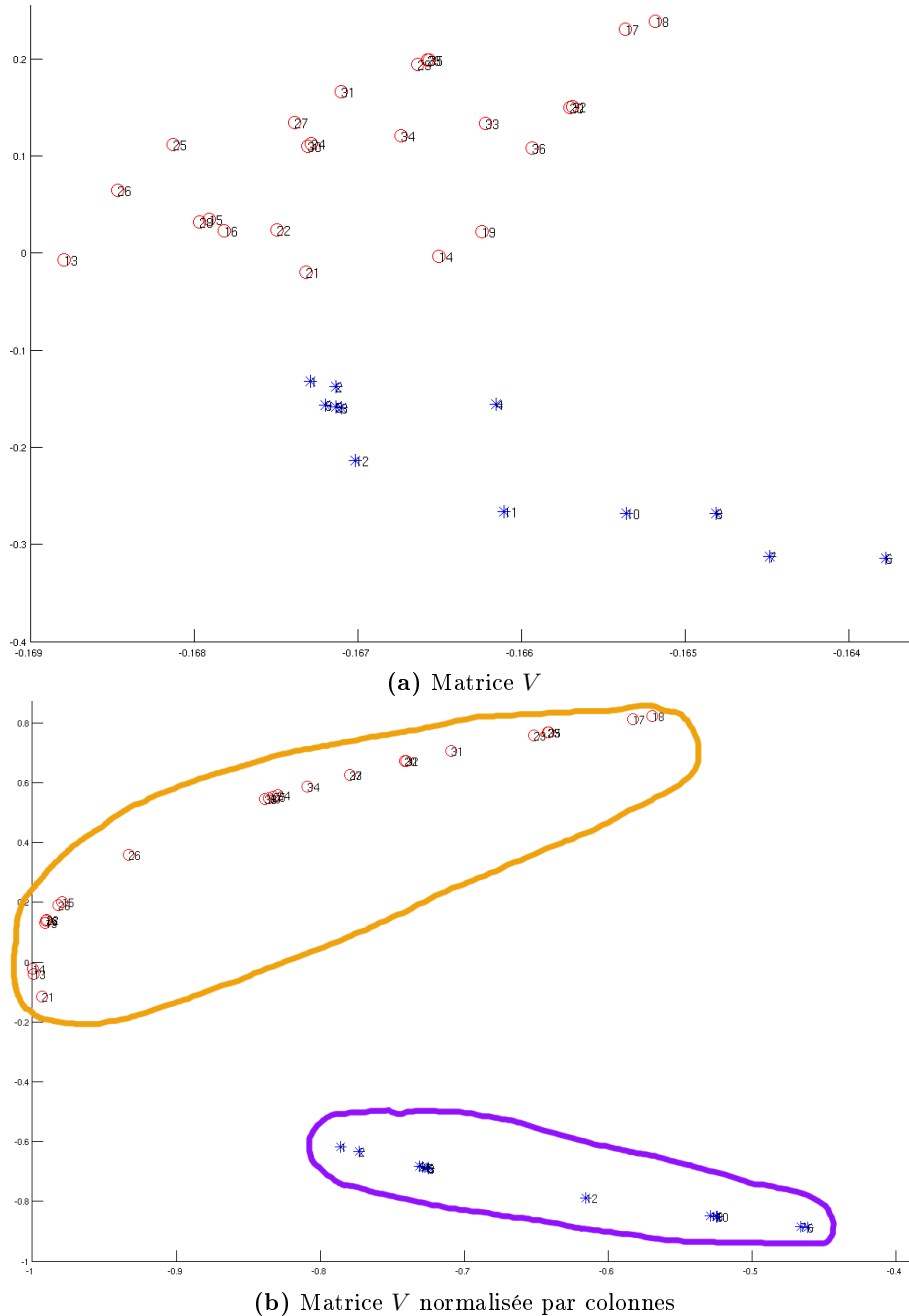


FIG. 42: Représentation des 36 courbes de la population GB (la couleur des points représente la vérité, en bleu la classe 1, en rouge la classe 2) dans le nouvel espace réduit à deux dimensions donc les coordonnées sont données par la matrice V des vecteurs propres retenus par la méthode. En (b), on voit le même résultat pour ces mêmes vecteurs propres normalisés. Les résultats du K -moyennes sont entourés de couleurs. On constate que toutes les courbes ont été bien classées.

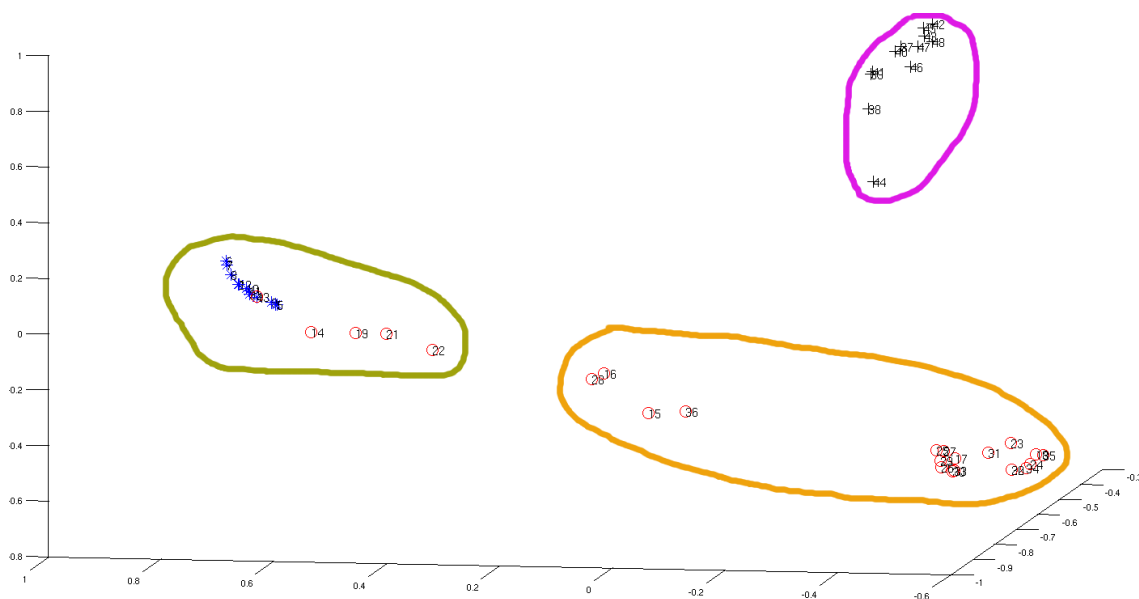
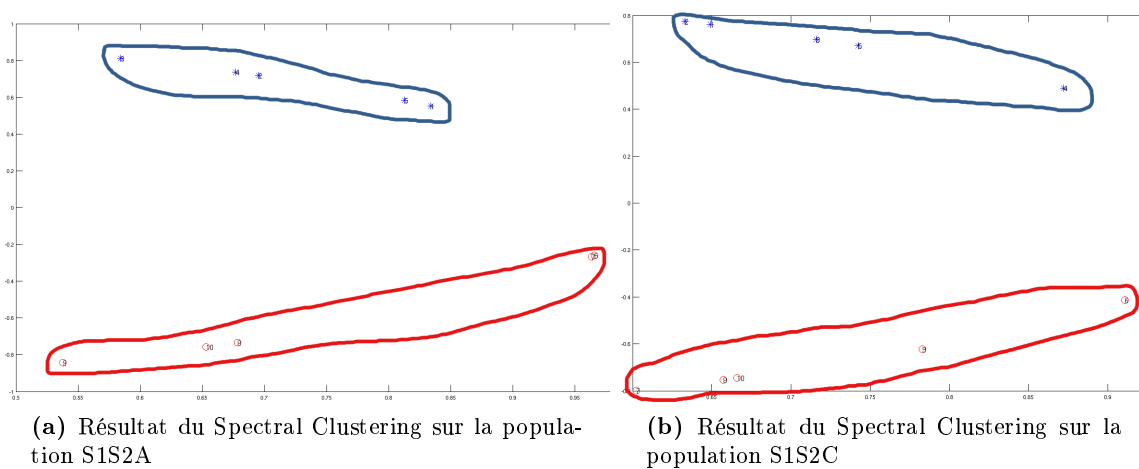


FIG. 43: 3 dimensions du Spectral Clustering sur la population de courbes GC comportant 3 classes. La couleur des points donne la (vraie) classe d'appartenance des points, les points noirs sont les courbes avec une bosse sur la droite, les points bleus sont les 12 courbes avec une bosse sur la gauche, et les 24 points rouges sont les courbes composées de deux gaussiennes. Ce qui est entouré est le résultat du 3-moyennes. On remarque que la classification n'est pas tout à fait exacte, car certaines courbes construites avec deux gaussiennes se retrouvent avec celles n'en ayant qu'une, mais ces courbes sont exactement les courbes pour lesquelles nous ne pouvons dire si elles ont vraiment une ou deux bosses, car les deux gaussiennes sont confondues (voir quelques unes de ces courbes à la figure 22)



(a) Résultat du Spectral Clustering sur la population S1S2A

(b) Résultat du Spectral Clustering sur la population S1S2C

FIG. 44: Résultats sur les populations d'hippocampes S1S2A et S1S2C. Les hippocampes sont correctement séparés.

7.2.3 Discussion

Le Spectral Clustering fonctionne plutôt bien quand les distances sont euclidiennes (figure 41) ou quand les données ne présentent pas de continuum entre les classes, comme c'est le cas pour les populations S1S2A et S1S2C. La classification des 3 classes de courbes de la population GC nous donne les groupes suivante :

- les « une bosse à gauche » et les « une bosse à gauche, presque deux bosses ».
- les « deux bosses »
- les « une bosse à droite »

Ce qui nous dérange en réalité dans ce résultat c'est que les « une bosse à gauche, presque deux bosses » soient classées avec les « une bosse à gauche » puisque par construction les presque deux bosses sont un mélange de gaussiennes, et donc même très proches, ces courbes ont en théorie deux bosses. Mais en regardant les déformations de ces formes ces mélanges de gaussiennes sont en fait plus proche de courbes n'ayant qu'une gaussienne. Le résultat n'est donc peut être pas si dérangeant que ça, il signifie que d'un point de vue de la forme, et non d'un point de vue construction mathématique, les « une bosse à gauche, presque deux bosses » n'ont pas deux bosses.

7.3 ISOMAP

Présentons maintenant la méthode ISOMAP introduite en 2000 par Tenenbaum et al [31] qui est une méthode de réduction de dimension non linéaire qui a la particularité de préserver la distance géodésique entre les individus, afin de tenir compte de la variété sur laquelle ils se trouvent.

7.3.1 Méthode

Premièrement, il faut construire la matrice d'adjacence du graphe. A partir de la matrice de distance, on seuille (comme pour le Spectral Clustering) soit en ne gardant que les n plus proches voisins de chaque sommets (ce qui a l'avantage de ne pas isoler de points), soit en seuillant la distance.

L'étape suivante consiste à estimer la matrice des distances géodésiques Dg entre les points en calculant le plus court chemin entre chaque sommet du graphe à l'aide d'algorithmes du type Dijkstra. On peut remarquer dès à présent un point important de la méthode : il est nécessaire d'avoir un certain nombre de points afin que le graphe soit assez dense si l'on souhaite obtenir une estimation de la distance géodésique correcte.

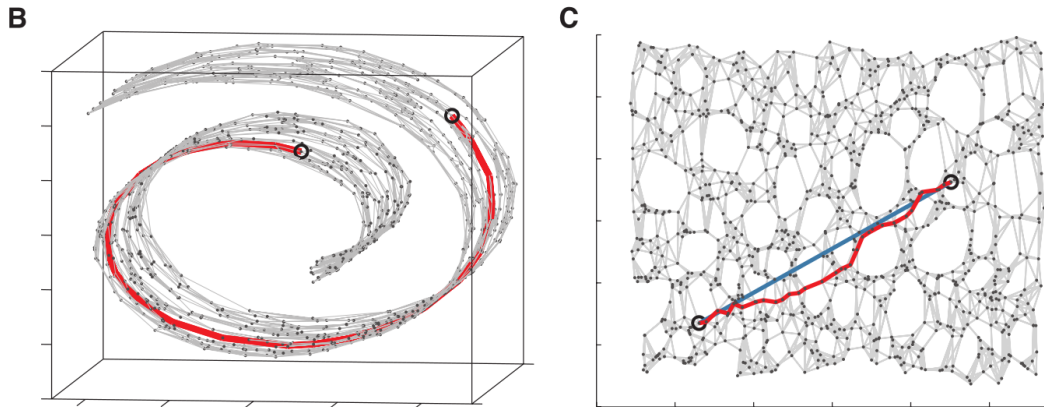


FIG. 45: Figures issues de l'article de Tenenbaum et al. 2000 [31]. En B, le graphe de la population des 1000 points (les sommets du graphe) pour lequel seul les 7 plus proches voisins ont été conservés, donne une approximation (en rouge) de la vraie distance géodésique. En C, Les points représentés en deux dimensions obtenues par ISOMAP. La ligne bleue représente alors une approximation plus simple et plus claire de la vraie distance géodésique que la ligne rouge qui représente le plus court chemin. Cette figure permet de confirmer, visuellement, la nécessité d'avoir beaucoup de points.

Maintenant on souhaite construire l'espace dans lequel sont les points : on calcule le carré de la matrice Dg noté Dg^2 , auquel on applique un double centrage : on a la matrice

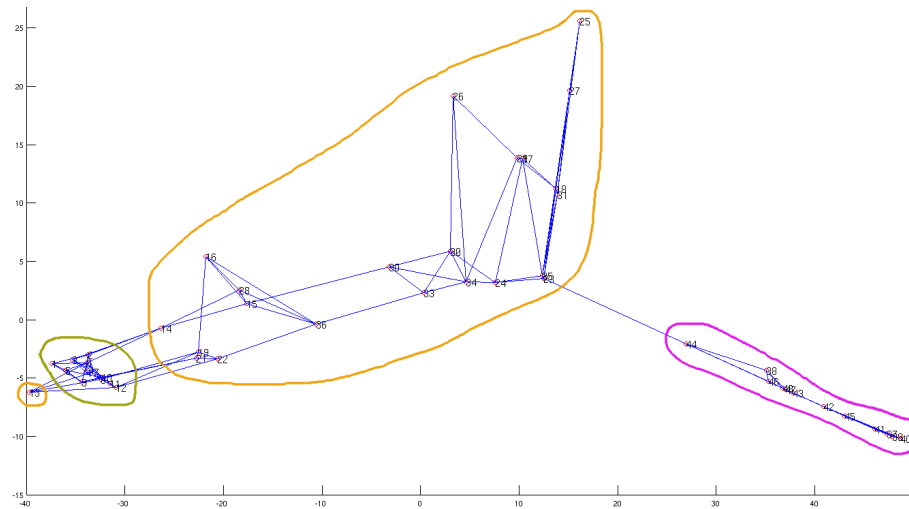
$$S = (Id - nbSuj^{-1}\mathbf{1}\mathbf{1}^t).1/2.Dg^2.(Id - nbSuj^{-1}\mathbf{1}\mathbf{1}^t)$$

. Il suffit ensuite d'extraire de la matrice S ses plus grandes valeurs propres et leurs vecteurs propres associés.

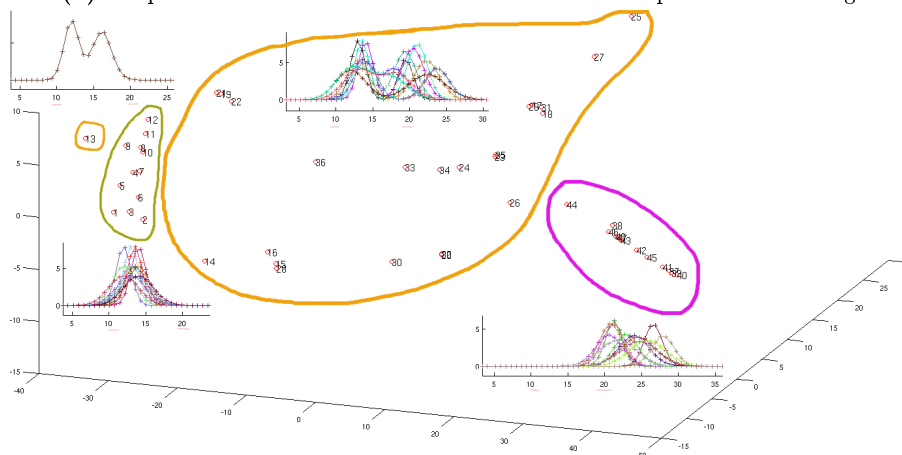
7.3.2 Résultats

Sur la figure 46, on voit que le graphe n'est pas aussi dense que pour une population contenant 1000 points (nous en avons ici 48). Mais il y a suffisamment de points pour remarquer la continuité entre les « une bosse à gauche » et les « deux bosses » du aux courbes ayant « presque deux bosses ». On remarque qu'une courbe, la numéro 13, est placée à gauche des « une bosse à gauche », alors qu'elle est composée de deux gaussiennes. On peut aussi remarquer sur l'image (a) de cette même figure que le premier axe de l'ISOMAP (axe des abscisses) suffirait à montrer la répartition des classes : à gauche

les « une bosse à gauche », au milieu, de façon assez étendue du fait de la plus grande variabilité que les deux autres classes nous avons les « deux bosses » et enfin à droite les « une bosse à droite ».



(a) Graphe résultat de l'ISOMAP en deux dimensions pour un 4 voisinage



(b) Résultat Sur les trois premières dimensions de l'ISOMAP

FIG. 46: Résultat de l'ISOMAP sur la population comportant 3 classes de courbes. (a), le graphe en deux dimensions (abscisses premier axe d'ISOMAP, en ordonnée on a le second axe), comme l'image droite de la figure 45. Chaque sommets à 4 voisins. (b), on peut voir les mêmes points sans les arcs du graphe en 3 dimensions. En jaune sont entourées les courbes à deux bosses, en rose les courbes avec une bosse sur la droite, et en vert les courbes avec 1 bosse sur la gauche.

7.3.3 Discussion

La méthode ISOMAP montre une répartition plus réaliste des données. En effet les individus dont la classe présente une relativement grande variabilité, sont représentés de manière plus dispersée que les autres ; comme la classe des deux bosses de la population de courbes GB qui comporte plus de variabilité (quasiement un continuum) que les deux autres classes de la population puisque les moyennes deux gaussiennes qui composent cette classe varient plus. La figure 45 le montre bien. Cette méthode, contrairement au Spectral Clustering, ne classe pas les individus. Visuellement nous pouvons avoir une idée du nombre de classes ou de la dispersion des individus (cf figure 45(a)). Comme vu dans l'état de l'art, nous pouvons faire le parallèle avec ce qu'on sait des données, dans notre cas nous savons où se trouvent les « une bosse » et « deux bosses », et nous pouvons ainsi dire que le résultat trouvé est logique car les « une bosse à gauche » sont à gauche, les « une bosse

à droite » sont à droite, et les « deux bosses » qui sont constitués de ces deux dernières classes, sont au milieu.

Comme nous l'avons dit plus haut, la méthode ne propose pas de classer les données, ce qui permet de choisir une méthode de classification. On remarque que dans le cas du résultat présenté à la figure 46, utiliser la méthode des k-moyennes pour classer les données ne donnerait pas le bon résultat. Par contre l'utilisation d'un algorithme comme le mean shift proposé en 1995 par Cheng [7], (algorithme utilisé dans la méthode atlas stratification, vue dans l'état de l'art) qui détecte de façon itérative des modes dans une population donnerait très certainement un bien meilleur résultat de classification, mais nous n'avons pas eu le temps de tester cette hypothèse.

Comme nous venons de le voir, le Spectral Clustering est aussi une bonne méthode de classification quand les distances des données sont euclidiennes (voir figure 41), faire un Spectral Clustering après ISOMAP ne paraît pas tout à fait absurde, puisque ISOMAP représente les données dont les distances sont géodésiques dans un nouvel espace dans lequel elles seront alors euclidiennes.

8 Conclusion

Le but de ce stage était de concevoir des méthodes pour analyser la variabilité anatomique au sein d'une population et en extraire des modèles représentants. Par rapport à d'autres méthodes proposant d'extraire d'une population les différents modes de variation, la particularité de ce stage est d'utiliser les grandes déformations difféomorphiques et de traiter du cas des surfaces et des courbes, plutôt que des images. Le contexte de ce travail est celui de l'imagerie de l'hippocampe par IRM 7T, qui permet d'accéder à sa structure interne, qui semble avoir une grande variabilité.

Durant ce stage, nous avons pu explorer trois types de méthodes qui identifient de façon très différentes les classes ou les tendances présentes au sein de la population.

La première est une méthode itérative qui répartit dans les classes les individus qu'elle juge appartenir vraiment à cette classe. Cette méthode ne partitionne pas la population, certains sujets ne sont pas attribués à une classe. Ce n'est pas forcément un problème dans notre cas, puisque l'on cherche à construire des modèles représentatifs et pas à classer la totalité de la population. L'avantage de cette méthode par rapport aux autres est qu'elle nous fournit directement les modèles de chaque classe. Mais elle présente des inconvénients importants : elle nécessite beaucoup de paramètres (la marge et le seuil pour la séparation des données, et K pour le nombre de classes) et est assez lente, nous n'avons d'ailleurs pas eu le temps de la tester sur des hippocampes.

La seconde approche était plus analytique, nous voulions alors voir comment se répartissaient les formes autour de leur modèle commun en analysant la répartition des vecteurs moments initiaux de la transformation partant de ce modèle vers les individus. Lorsque les classes sont bien séparées, nous pouvons visuellement bien les distinguer, et ainsi évaluer le nombre de classes qui nous fait souvent défaut. Dans le cas où les classes ne sont pas bien séparées, nous ne pouvons estimer visuellement le nombre de classes, mais on a pu voir que des tirs géodésiques le long de l'axe principal permettent de bien décrire les tendances générales de la population. Cette méthode permet d'avoir une idée un peu plus précise sur la répartition des données de la population. Par contre, elle ne fournit pas les différents modèles présents dans la population. C'est peut-être une méthode à utiliser pour une analyse préliminaire, car elle est en plus peu coûteuse en temps, avant de chercher à estimer les modèles proprement dits.

La dernière approche propose de visualiser les données dans un autre espace en se basant sur la matrice regroupant toutes les distances inter-individu. Le calcul exact serait très coûteux mais nous avons vu que nous pouvions approcher cette distance. Ces méthodes basées sur la similarité entre les individus partitionnent les individus dans des classes cohérentes. Après ce partitionnement des individus, il est possible de calculer un modèle par classe afin d'avoir tous les modèles représentants de la population. L'une de ces méthodes a l'inconvénient d'avoir comme paramètre le nombre K de classes, comme la première approche, mais contrairement à cette dernière il est facile de tester plusieurs valeurs de K car les résultats sont immédiats. L'autre, ISOMAP, ne demande pas de connaître le nombre de classes, et en plus propose une représentation des données dans un espace plus facile à exploiter que l'ACP, surtout en cas de continuum entre certaines classes. Le problème de l'ISOMAP est qu'il requiert un grand nombre de sujets. Par ailleurs, il ne partitionne pas les individus ni ne les classe. Un algorithme de classement est ensuite nécessaire, comme par exemple le spectral clustering ou le K-moyennes adapté.

*

**

Lors de ce stage, un assez grand nombre de pistes ont été explorées. On a privilégié l'exploration de différentes stratégies plutôt que de faire une analyse très détaillée d'une seule d'entre elles pour différentes raisons. Tout d'abord, à cause du nombre très réduit de données qui nous empêchaient de faire une validation complète. Par ailleurs, je vais poursuivre en thèse, il était intéressant d'ouvrir différentes pistes pour pouvoir ensuite guider les choix de méthodes qui seront développées durant la thèse.

Le but de cette thèse est de proposer des méthodes pour étudier la variabilité anatomique de l'hippocampe et du système limbique dont il fait partie et qui est notamment impliqué dans la maladie d'Alzheimer.

Contrairement au cas du stage, nous ne ferons pas a priori sur le fait que la population a plusieurs modes et nous regarderons la variabilité anatomique à la fois du point de vue continu et du point de vue discret. Pour la création de ces méthodes nous utiliserons plutôt des méthodes comme ISOMAP pour la continuité ou Spectral Clustering pour le cas discret, car le stage nous a permis de voir que ces méthodes étaient les plus adaptées quand la population est grande. Nous pourrions commencer par analyser les données avec des méthodes comme l'ACP à noyau et/ou l'ISOMAP afin de déterminer si la variabilité est plutôt continue ou discrète et dans ce dernier cas d'estimer le nombre de classes. Le spectral clustering pourrait donner une première approximation du représentant de chaque classe, et ainsi initialiser correctement le K-moyennes adapté, qui une fois bien initialisé converge en peu d'itérations et donne des modèles assez purs, constitués que par des formes appartenant « clairement » à la classe.

Ces travaux seront appliqués à l'étude des variations morphologiques de l'hippocampe et d'autres structures voisines (comme l'amygdale, le sillon collatéral, le sillon cingulaire...) et à l'étude de l'impact de certain gènes (notamment ceux connus comme facteurs de risque pour la maladie d'Alzheimer) sur cette variabilité. Nous savons déjà que l'un d'eux est responsable d'une configuration particulière du cortex entorhinal, cortex qui fait partie du système limbique et qui constitue la principale entrée de l'hippocampe et est important dans la consolidation de la mémoire.

Nous avons pu voir au cours du stage que certaines des méthodes ne sont vraiment intéressantes que lorsqu'on a de nombreux sujets. Pour commencer, nous travaillerons sur des hippocampes acquis à 3T dont de grandes bases de données segmentées existent au laboratoire. On pourra envisager en fin de thèse de travailler à nouveau sur des données 7T si un plus grand nombre d'entre elles sont disponibles, ce qui nécessite par ailleurs le développement d'une méthode de segmentation automatique de l'hippocampe à 7T (ce travail est en cours dans le cadre du postdoc de Linda Marrakchi).

Références

- [1] S. Allasonnière, E. Kuhn, and A. Trouvé. MAP estimation of statistical deformable templates via nonlinear mixed effects models : Deterministic and stochastic approaches. In *Proc. of the International Workshop on the Mathematical Foundations of Computational Anatomy (MFCA-2008)*, September 2008.
- [2] G. Auzias. *Recalage interindividuel de surfaces corticales par déformations difféomorphiques*. PhD thesis, Université Paris-Sud 11, 2009.
- [3] G. Auzias, J. Glaunes, O. Colliot, M. Perrot, J.-F. Mangin, A. Trouvé, and S. Baillet. DISCO : A coherent diffeomorphic framework for brain registration under exhaustive sulcal constraints. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009*, pages 730–738, 2009.
- [4] D. J. Blezek and J. V. Miller. Atlas stratification. *Medical Image Analysis*, 11(5) :443–457, 2007.
- [5] F.L. Bookstein. Principal warps : Thin-plate splines and the decomposition of deformations. *IEEE Transactions on, Pattern Analysis and Machine Intelligence*, 11(6) :567–585, 1989.
- [6] E. Ceyhan, M. Beg, C. Ceritoglu, L. Wang, J. Morris, J. Csernansky, M.I. Miller, and T. Ratananather. Quantization and analysis of hippocampal morphometric changes due to dementia of Alzheimer type using metric distances based on large deformation diffeomorphic metric mapping. *Computerized Medical Imaging and Graphics : The Official Journal of the Computerized Medical Imaging Society*, 35(4) :275–293, June 2011.
- [7] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(8) :790–799, 1995.
- [8] M. Chupin, S. Lehericy, D. Hasboun, O. Colliot, U. Goerke, M. Marjanska, K. Ugurbil, and P-F van der Moortele. Segmenting the subregions of the human hippocampus at 7 Tesla. *NeuroImage*, 47(Supplement 1) :S122, 2009. Organization for Human Brain Mapping Annual Meeting.
- [9] D.L. Collins. *Model-based segmentation of individual brain structures from magnetic resonance imaging data*. PhD thesis, McGill University, 1994.
- [10] G. Davis., S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Constructive Approximation*, 13(1) :57–98, March 1997.
- [11] S. Durrleman, X. Pennec, A. Trouvé, and N. Ayache. Statistical models of sets of curves and surfaces based on currents. *Medical Image Analysis*, 13(5) :793–808, 2009.
- [12] H. Duvernoy. *The Human Hippocampus : An Atlas of Applied Anatomy*. Bergmann Verlag, Munich, 1988.
- [13] A.C. Evans, D.L. Collins, S. Mills, E. Brown, R. Kelly, and T. Peters. 3D statistical neuroanatomical models from 305 MRI volumes. In *Nuclear Science Symposium and Medical Imaging Conference, 1993*, pages 1813–1817. IEEE, 1993.
- [14] J. Glaunes. *Transport par difféomorphismes de points, de mesures et de courants pour la comparaison de formes et l'anatomie numérique*. PhD thesis, Université Paris 13, 2005.
- [15] J. Glaunès and S. Joshi. Template estimation from unlabeled point set data and surfaces for computational anatomy. In X. Pennec and S. Joshi, editors, *Proc. of the International Workshop on the Mathematical Foundations of Computational Anatomy (MFCA-2006)*, pages 29–39, 1st of October 2006.
- [16] J. Glaunes, A. Qiu, M. I. Miller, and L. Younes. Large deformation diffeomorphic metric curve mapping. *International Journal of Computer Vision*, 80(3) :317–336, 2008.
- [17] J. Glaunes, A. Trouvé, and L. Younes. Diffeomorphic matching of distributions : A new approach for unlabelled point-sets and sub-manifolds matching. *Computer Vision and Pattern Recognition*, pages 712–718, 2004.

- [18] S. Joshi and M. Miller. Landmark matching via large deformation diffeomorphisms. *IEEE Transactions on Signal Processing*, 9(8) :1357–1370, 2000.
- [19] R. Kothari and D. Pitts. On finding the number of clusters. *Pattern Recognition Letters*, 20(4) :405–416, April 1999.
- [20] P. Lorenzen, B. Davis, and S. C. Joshi. Unbiased atlas formation via large deformations metric mapping. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005*, pages 411–418, 2005.
- [21] J.F. Lu, J.B. Tang, Z.M. Tang, and J.Y. Yang. Hierarchical initialization approach for K-means clustering. *Pattern Recognition Letters*, 29(6) :787–795, April 2008.
- [22] L. Van Der Maaten, E. Postma, and J. Van Den Herik. Dimensionality reduction : A comparative review. *Tilburg University Technical Report*, 10 :35, 2009.
- [23] S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41 :3397–3415, 1993.
- [24] J. Mazziotta, A. Toga, A. Evans, P. Fox, and J. Lancaster. A probabilistic atlas of the human brain : theory and rationale for its development : International consortium for brain mapping (ICBM). *NeuroImage*, 2(2PA) :89–101, 1995.
- [25] J. Mazziotta, A. Toga, A. Evans, P. Fox, J. Lancaster, K. Zilles, and R. Woods. A probabilistic atlas and reference system for the human brain : International consortium for brain mapping (ICBM). *Philosophical Transactions : Biological Sciences*, 356(1412), 2001.
- [26] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering : Analysis and an algorithm. In *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 14 : Proceeding of the 2001 Conference*, pages 849–856. MIT Press, 2001.
- [27] D. Pelleg and A. W. Moore. X-means : Extending k-means with efficient estimation of the number of clusters. *The International Conference on Machine Learning (ICML)*, pages 727–734, 2000.
- [28] Z. Yi Sun, M. Perrot, A. Tucholka, D. Rivière, and J.-F. Mangin. Constructing a dictionary of human brain folding patterns. In *MICCAI*, pages 117–124, 2009.
- [29] Z.Y. Sun. *Inférence d'un dictionnaire des motifs des plissements corticaux*. PhD thesis, Université Paris-Sud 11, 2011.
- [30] J. Talairach and P. Tournoux. Co-planar stereotactic atlas of the human brain : 3-dimensional proportional system : an approach to cerebral imaging. *Thieme Medical Publishers*, page 122, 1988.
- [31] J. Tenenbaum, V. Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500) :2319–2323, December 2000.
- [32] L. Wang, F. Beg, T. Ratnanather, C. Ceritoglu, L. Younes, J. Morris, J. Csernansky, and M. I. Miller. Large deformation diffeomorphism and momentum based hippocampal shape discrimination in dementia of the Alzheimer type. *IEEE Transactions on Medical Imaging*, 26(4) :462–470, April 2007.
- [33] X. F. Yang, A. Goh, and A. Qiu. Approximations of the diffeomorphic metric and their applications in shape learning. In *Information Processing in Medical Imaging :IPMI*, pages 257–270, 2011.