

ACADÉMIE DE MONTPELLIER
UNIVERSITÉ MONTPELLIER II
Sciences et Techniques du Languedoc

THÈSE

présentée au Laboratoire d'Informatique de Robotique
et de Microélectronique de Montpellier pour
obtenir le diplôme de doctorat

Spécialité : **Informatique**
Formation Doctorale : **Informatique**
École Doctorale : **Information, Structures, Systèmes**

Une approche combinatoire novatrice fondée sur les matroïdes orientés pour la caractérisation de la morphologie 3D des structures anatomiques

par

Kevin SOL

12 septembre 2013

Directeur de thèse

M. Emeric GIOAN Chargé de Recherche CNRS, LIRMM, Université Montpellier II

Co-Directeur de thèse

M. Gérard SUBSOL Chargé de Recherche CNRS, LIRMM, Université Montpellier II

Rapporteurs

M. Xavier PENNEC Directeur de Recherche INRIA, Sophia Antipolis

M. Michel POCCHIOLA Professeur, IMJ, Université Paris 6

Examineur

M. José BRAGA Professeur, AMIS, Université Toulouse 3

M. Jorge RAMÍREZ ALFONSÍN Professeur, I3M, Université Montpellier II



Table des matières

Table des matières	i
Introduction	1
Partie I. Préliminaires et motivations	5
1 Techniques de morphométrie 3D en anthropologie	7
1.1 Définition des repères ou landmarks	7
1.2 Objectifs de la morphométrie	9
1.3 Méthodes de morphométrie 3D à base de landmarks utilisées en anthropologie . . .	10
1.3.1 Les méthodes “traditionnelles”	11
1.3.2 Méthodes par superposition	11
1.3.3 Méthodes par déformation	14
1.3.4 Euclidean Distance Matrix Analysis	15
1.4 Limites de ces méthodes	17
2 Codage d’un modèle 3D par un matroïde orienté	21
2.1 Comment associer un matroïde orienté à un ensemble de points 3D?	22
2.1.1 Définition du matroïde orienté associé à un ensemble de points	22
2.1.2 Exemples de matroïdes orientés associés à des configurations de points en deux dimensions	26
2.1.3 Exemples de matroïdes orientés associés à des configurations de points en trois dimensions	26
2.1.4 Quelques définitions équivalentes du matroïde orientés associé à un ensemble de points	28
2.2 Propriétés géométriques codées par les matroïdes orientées	32
2.3 Introduction d’une distance entre matroïdes orientés	33
2.4 Avantages et limites pour le codage d’ensembles de points	35

Partie II. Bases fixes issues d'ordres sur les coordonnées des points	37
3 Characterization in dimension 2 and 3 and conjecture in higher dimensions	39
3.1 Preliminaries	41
3.1.1 Formalism and terminology of the problem	41
3.1.2 An example from applications	42
3.2 Computable fixity criteria and conjectures	44
3.2.1 From partial orderings to linear orderings	44
3.2.2 Formal fixity	45
3.2.3 Formal fixity by expansion	47
3.2.4 A non-fixity criterion	48
3.3 Characterizations in low dimensions	49
3.3.1 Results in dimension 2	49
3.3.2 Results in dimension 3	51
3.3.3 Example continued	53
3.3.4 Proofs of Theorem 3.10 and Theorem 3.11	54
4 Calcul du nombre d'ensembles d'ordres totaux à transformations près	63
4.1 Problématique générale	63
4.2 Un encadrement	65
4.2.1 Une borne supérieure	65
4.2.2 Une borne inférieure	68
4.3 Méthode de calcul de la valeur exacte	69
4.3.1 Recherche d'un ensemble d'ordres totaux équivalent	69
4.3.2 Algorithmes pour calculer la valeur exacte	73
4.3.3 Comparaison entre la valeur exacte et son encadrement	76
Partie III. Classification et caractérisation de chirotopes	77
5 Analyse de méthodes de classification existantes	81
5.1 Transformer un chirotope en données de différents types	81
5.1.1 Transformation en données binaires	82
5.1.2 Transformation en données numériques	84
5.2 Méthodes de clustering	87
5.2.1 Classification ascendante hiérarchique	88
5.2.2 k -means	93
5.2.3 k -medoids	97
5.2.4 Analyse d'autres méthodes de clustering	101
5.3 Méthodes de classification supervisée	102
5.3.1 Arbres de décision	103
5.3.2 Classification naïve bayésienne	108
5.3.3 Réseaux de neurones artificiels	109
6 Proposition d'une nouvelle méthode de caractérisation	111

6.1	Cas uniforme	111
6.2	Cas non-uniforme	114
6.2.1	Définitions et exemples	114
6.2.2	Caractérisations faible et forte des classes	119
6.3	Retour sur le cas uniforme	121
6.4	Liens entre les taux de caractérisation et de discriminabilité	123
6.5	Calcul des paramètres de caractérisations faibles et fortes	124
6.6	Exemple récapitulatif	126
Partie IV. Implémentation logicielle		131
7	Implémentation du calcul du chirotope	133
7.1	Matroïde orienté associé à une configuration de points en dimension quelconque . .	133
7.1.1	Méthodes de calcul du chirotope, des cocircuits et des circuits	133
7.1.2	Exactitude des calculs	136
7.1.3	Complexité théorique et temps de calcul	136
7.2	Calcul de l'indice d'un $(d + 1)$ -uplet	137
7.3	Réflexions sur les fichiers de données et de résultats	138
8	Implémentation du reste de la méthode	141
8.1	Calcul de l'ensemble des bases fixes issues d'ordres sur les coordonnées des points .	141
8.2	Caractérisation forte de classes	144
8.3	Calcul des symétries et asymétries	145
Partie V. Applications à la caractérisation de la morphologie 3D		147
9	Anatomie comparée : analyse des différences humains / grands singes	151
9.1	Calcul des chirotopes et distances entre les échantillons	152
9.2	Classification automatique	153
9.3	Caractérisation des différentes espèces	154
10	Orthodontie : analyse de classes dentaires	159
11	Déformation cranio-faciale : analyse de craniosynostoses	165
11.1	Etude sur l'ensemble des landmarks	167
11.2	Etude en utilisant uniquement les landmarks anatomiques	170
11.3	Etude de la symétrie des crânes	176
Conclusion et perspectives de recherche		189
Bibliographie		195
Index		203
Table des figures		205



Introduction

Thématiques de la thèse

Dans cette thèse, nous abordons une question fondamentale en anatomie : *comment quantifier la différence de forme de structures anatomiques ?* C'est un sujet qui a de nombreuses applications tant pour comprendre notre corps (par exemple, quelle est la variabilité de la mandibule chez les enfants entre 3 et 5 ans ?) et son évolution (quelle est la différence entre un crâne d'Australopithecus africanus et d'Homo sapiens ?) que pour la pratique médicale (comment caractériser une structure pathologique, comment suivre l'évolution lors d'un traitement thérapeutique ?). Pour cela, nous proposons une approche combinatoire novatrice fondée sur un outil mathématique : *les matroïdes orientés* qui nous permettra notamment de détecter les différences "structurelles" sans utiliser les notions de distances ou d'angles.

Cette thèse s'articule donc autour de deux thèmes : l'étude quantitative de formes (aussi appelée morphométrie) et les matroïdes orientés.

Morphométrie. La morphométrie est l'étude et l'analyse quantitative de la forme des objets. Elle est en particulier utilisée en anatomie et en anthropologie.

De nombreuses approches sont basées sur des points de repère (appelés landmarks en anglais), qui sont en général des points ayant une signification anatomique. Les experts relèvent sur chaque échantillon les coordonnées des mêmes points de repère afin de les comparer.

Actuellement, l'analyse de ces différences de formes se fonde sur l'étude des distances ou des angles entre des points de repère (par exemple, les mesures craniométriques) ou de paramètres métriques caractérisant la déformation entre des configurations de points de repère relevés sur plusieurs échantillons (voir la méthode de "morphométrie géométrique").

Or il arrive que les différences significatives (c'est-à-dire qui ne dépendent pas que de la variabilité interindividuelle normale) entre des structures anatomiques ne soient pas métriques mais "structurelles". Par exemple, dans le cas d'une face prognathe, c'est tout un sous-ensemble de points de repère qui passe devant un autre, de manière groupée et corrélée. Ceci n'est pas toujours bien

mis en évidence par les variations des coordonnées des points de repère, ce qui suggère d'introduire de nouveaux outils de morphométrie 3D.

Matroïdes orientés. La théorie des matroïdes orientés est développée depuis une quarantaine d'années dans le cadre de recherches théoriques en algèbre linéaire et en théorie des graphes. Cette théorie fournit un cadre général permettant de modéliser, de décrire et d'analyser les propriétés combinatoires (c'est-à-dire de façon non numérique) de configurations géométriques. Des objets mathématiques apparemment totalement différents tels que des configurations de points ou de vecteurs, des arrangements d'hyperplans, des polytopes convexes, des graphes orientés, et des programmes linéaires trouvent une généralisation commune dans la langue de matroïdes orientés.

Un matroïde orienté nous permet de coder la position relative des points de façon purement combinatoire, c'est-à-dire sans utiliser de notions d'angles ou de distances, en associant un signe (0, + ou -) à chaque sous-ensemble de $(d+1)$ points où d est la dimension de l'espace (dans notre cas 2 ou 3). Le signe associé à un ensemble de $(d+1)$ points correspond à l'orientation du polytope formé par ces points. Le vecteur de signes est appelé *chirotope*. Le chirotope semble donc être un outil naturel pour détecter les différences "structurelles" entre les différents échantillons.

Dans cette thèse, nous proposons de modéliser les configurations de points de repère 3D par des matroïdes orientés. Nous verrons que cette modélisation nous permettra de détecter des changements structurels comme le passage d'un point de repère à travers le plan formé par trois autres.

Problèmes étudiés et plan de la thèse

Dans la Partie I nous introduisons les deux grands thèmes de cette thèse : la morphométrie 3D et les matroïdes orientés. Dans le Chapitre 1 nous présentons différents types de points de repère et les méthodes utilisées par les experts pour les relever sur des structures anatomiques. Nous décrivons ensuite les objectifs de la morphométrie puis nous présentons quelques méthodes utilisées en anthropologie pour l'étude de structures anatomiques en deux ou trois dimensions. Nous aborderons enfin les limites de ces méthodes. Dans le Chapitre 2, nous présentons comment un matroïde orienté nous permet de coder la position relative de points en associant un signe (0, + ou -) à chaque sous-ensemble de $(d+1)$ points (où d est la dimension de l'espace). Nous expliquons ensuite l'apport qu'il peut avoir pour la description de formes en deux ou trois dimensions.

Dans la Partie II, nous nous intéressons aux ensembles de $(d+1)$ points dont l'orientation est la même pour tous les échantillons. En fait, nous travaillons avec des points de repère anatomiques ce qui implique implicitement des relations que nous pouvons formaliser par des contraintes d'ordre sur leurs coordonnées. Par exemple, si l'on considère les deux oreilles internes, la pointe du menton et le milieu du front d'un crâne alors l'oreille interne droite sera le point le plus à droite et l'oreille interne gauche sera le point le plus à gauche, et ce, quel que soit le crâne et la variabilité naturelle. La question qui se pose alors est : *est-ce que ces contraintes d'ordre impliquent que le signe associé à ces quatre points est le même quel que soit le crâne étudié ?* Dans cette partie nous nous intéressons donc à des simplexes en dimension quelconque pour lesquels, pour chaque coordonnée de l'espace, un ordre est fixé entre les différents sommets. Dans le Chapitre 3, nous cherchons à déterminer l'orientation de tels simplexes uniquement à partir des ordres donnés. Nous proposons un algorithme combinatoire en dimension deux et trois, qui à partir d'un ensemble d'ordres

sur les coordonnées, détermine si l'orientation d'un simplexe est constante quelles que soient les coordonnées des points.

Dans le Chapitre 4, nous nous focalisons sur un problème particulier de la méthode qui est de trouver, en fonction de la dimension, le nombre d'ensembles d'ordres totaux à équivalence près. Nous expliquons comment fonctionne le programme qui nous a permis de calculer ce nombre pour des petites dimensions.

Dans la Partie III, nous nous intéressons cette fois-ci aux simplexes dont l'orientation n'est pas la même pour tous les échantillons. Nous essayons de répondre aux deux grands objectifs de la morphométrie que sont la classification et de la caractérisation de classes. Dans le Chapitre 5, nous étudions certaines méthodes de classification déjà existantes afin de déterminer si elles peuvent être appliquées à un ensemble de chirotopes et si elles peuvent nous fournir une caractérisation des classes. Nous proposons ensuite dans le Chapitre 6 notre propre méthode de caractérisation, que nous décrivons d'abord dans le cas où les points sont en position générale puis dans le cas quelconque. Ces caractérisations permettent de définir chaque classe à l'aide d'une liste de sous-ensemble de $(d+1)$ points, d'un vecteur de signes et d'un entier.

Dans la Partie IV, nous présentons les problèmes théoriques et pratiques que nous avons résolus pour programmer les algorithmes présentés dans les trois premières parties.

Le Chapitre 7 est consacré au calcul du chirotope. D'abord, nous présentons l'algorithme permettant de calculer le chirotope d'une configuration de points en dimension quelconque. Nous expliquons ensuite la méthode employée pour garantir l'exactitude des calculs et nous réalisons une étude de la complexité des algorithmes. Comme le chirotope d'une configuration de points est un vecteur de signes, nous présentons en particulier comment calculer dans ce vecteur, la position du signe d'un ensemble de $(d+1)$ points. Enfin, nous expliquons les choix qui ont été fait concernant le format des fichiers de données et des fichiers retournés par le programme.

Le Chapitre 8 est consacré aux problèmes théoriques et pratiques qui se sont posés lors de la programmation des autres étapes de notre méthode. D'abord, nous présentons le programme en dimension deux et trois, qui à partir d'un ensemble d'ordres sur les coordonnées des points, détermine si l'orientation d'un simplexe est constante quelles que soient les coordonnées des points. Puis nous présentons un algorithme qui calcule les paramètres définies dans le Chapitre 5 nous permettant d'avoir une caractérisation de classes de chirotopes. Enfin nous étudions la symétrie (et l'asymétrie) d'une configuration de points de \mathbb{R}^d et d'un hyperplan de \mathbb{R}^d . En particulier nous présentons un programme permettant à partir du chirotope de la configuration de points, d'extraire le chirotope d'une demi-structure et de calculer le chirotope du symétrique de l'autre demi-structure par rapport à l'hyperplan.

Pour conclure cette thèse, nous présentons dans la Partie V les résultats obtenus en appliquant notre méthode à des jeux de données anatomiques. La première application (Chapitre 9) concerne l'anatomie comparée puisqu'il s'agit d'étudier la forme de crânes d'humains et de grands singes (bonobos et chimpanzés). Les deux autres applications se font sur des jeux de données cliniques. Il s'agit d'étudier respectivement les classes dentaires utilisées en orthodontie (Chapitre 10) et les déformations cranio-faciales présents chez des sujets atteints de craniosynostoses (Chapitre 11). Dans ce dernier chapitre nous étudions également la symétrie des crânes.

Publications et conférences

Les travaux réalisés dans cette thèse ont donné lieu à plusieurs publications dans des conférences avec comité de lecture et une soumission dans une revue internationale.

Conférences internationales avec actes.

- [23] *A combinatorial method for 3D landmark-based morphometry : application to the study of coronal craniosynostosis*. Avec Emeric Gioan et Gérard Subsol. **15th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2012)**, Nice (France) October 2012, Lecture Notes in Computer Science 7512, p. 533-541, Springer
- [21] *Orientations of Simplices Determined by Orderings on the Coordinates of their Vertices (CCCG'12)*. Avec Emeric Gioan et Gérard Subsol. **23rd Canadian Conference on Computational Geometry**, Toronto (Canada), August 2011. 187-192, p. 187-192

Conférence internationale avec résumé.

- [24] *A new 3D morphometric method based on a combinatorial encoding of 3D point configurations : application to skull anatomy for clinical research and physical anthropology*. Avec Emeric Gioan, Gérard Subsol, Yann Heuzé, Joan Richtsmeier, José Braga et Francis Thackeray. **80th Annual Meeting of the American Association of Physical Anthropologists** (2011) Résumé publié dans : American Journal of Physical Anthropology, p. 280, Vol. 144 Issue S52

Conférence nationale avec résumé.

- [25] *Une nouvelle méthode de morphométrie 3D par codage combinatoire de configurations de points 3D : application à l'anatomie du crâne*. Avec Emeric Gioan, Gérard Subsol, José Braga, Jacques Treil. **1836èmes Journées de la Société d'Anthropologie de Paris**(2011) Résumé publié dans : Bulletins et Mémoires de la Société Anthropologique de Paris, Vol. 23, Suppl. 1, p. S35

Soumission à une revue internationale.

- *Orientations of Simplices Determined by Orderings on the Coordinates of their Vertices*. Avec Emeric Gioan, Gérard Subsol. Soumis à **Linear Algebra and its Applications** en mai 2013, 24 pages



Partie I. Préliminaires et motivations

Dans cette partie nous introduisons les deux grands thèmes de cette thèse : la morphométrie 3D et les matroïdes orientés. Dans le Chapitre 1 nous décrivons les objectifs de la morphométrie puis nous présentons quelques méthodes utilisées en anthropologie pour l'étude de structures anatomiques en trois dimensions, puis nous abordons les limites de ces méthodes. Dans le Chapitre 2, nous présentons brièvement ce que sont les matroïdes orientés et nous expliquons l'apport qu'ils peuvent avoir pour la description de formes en trois dimensions.

Techniques de morphométrie 3D en anthropologie

La morphométrie est l'étude et l'analyse quantitative de la forme des objets. Elle est en particulier utilisée en anatomie et en anthropologie. On dispose d'un ensemble d'échantillons, par exemple un ensemble de crânes, dont on souhaite quantifier les différences de forme.

De nombreuses approches sont basées sur des points de repères ou amers appelés landmarks en anglais (voir Section 1.1), qui sont en général des points ayant une signification anatomique. Les experts relèvent sur chaque échantillon les mêmes landmarks, puis les différences de formes des échantillons sont étudiées en utilisant les coordonnées en trois dimensions de ces landmarks.

D'autres approches se basent sur des primitives géométriques plus complexes comme des droites, des plans, des normales ou des repères. Ces approches utilisent des outils mathématiques comme les variétés différentielles ou les groupes de Lie pour étudier les différences de forme.

1.1 Définition des repères ou landmarks

Le principe des méthodes de morphométrie se basant sur des landmarks est de discrétiser la forme, c'est-à-dire que l'analyse ne se fait pas sur tout le contour ou la surface de la structure mais sur un nombre fini de points. Pour que l'on puisse comparer la forme des structures entre chaque échantillon, il faut qu'il y ait une correspondance entre les points des échantillons. Ainsi si l'on souhaite comparer des visages, on peut prendre comme landmark le coin des yeux ou la pointe du menton mais ni un grain de beauté ni l'extrémité d'une ride d'expression puisque ces derniers ne seront certainement pas présents pour tous les visages.

On peut distinguer trois types de landmarks [7, 52]. Le premier type de landmarks est constitué des *landmarks anatomiques*, c'est-à-dire de points qui ont une signification biologique, comme par exemple la jonction de deux os ou un foramen (trou permettant de laisser passer un muscle, un nerf, ou un vaisseau sanguin). Ces landmarks sont clairement définis du point de vue anatomique et peuvent être facilement situés (et donc relevés) pour chaque échantillon, ce qui nous assure que les landmarks anatomiques sont en correspondance dans les différents échantillons. Par contre la localisation de ces landmarks peut être subjective [75], c'est-à-dire que deux experts ne les place-

landmarks
anato-
miques

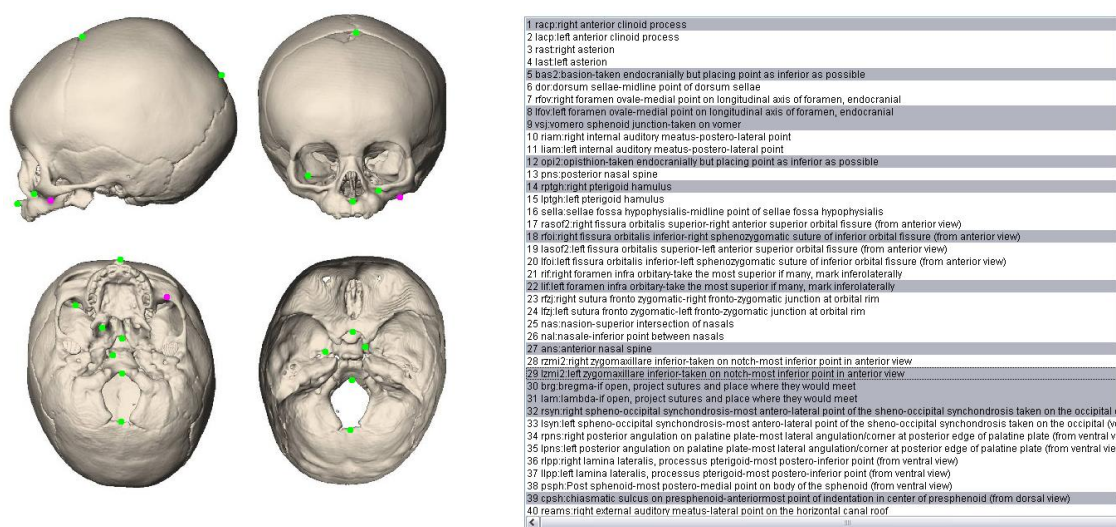


FIGURE 1.1 – Exemples de landmarks anatomiques relevés sur un crâne humain. Cette figure a été faite grâce à une des applications du site web http://getahead.psu.edu/landmarks_new.html

raient peut-être pas exactement au même endroit. De plus la localisation d'un landmark anatomique peut varier beaucoup d'un individu à un autre sans que cette modification soit considérée comme pathologique (c'est le cas par exemple pour la jonction des sutures du crâne). La Figure 1.1 représente un exemple de landmarks anatomiques. Cette figure a été faite grâce à une application du site web du "Richtsmeier lab" http://getahead.psu.edu/landmarks_new.html, laboratoire faisant parti du Department of Anthropology de la Pennsylvania State University.

Les deux autres types de landmarks ont une signification géométrique. Le deuxième type de landmarks est constitué de landmarks indiquant des "dépressions" ou des "sommets" dans la structure, ce qui correspond à des minima locaux et à des maxima locaux de la courbure. Par exemple, la pointe d'une canine ou le creux d'une molaire sont des landmarks de ce type. Le troisième type de landmarks est composé de points extrémaux dans une direction de l'espace, comme par exemple le sommet du crâne. Le problème des landmarks du troisième type est qu'ils ne peuvent pas être situés de façon précise puisqu'ils ne correspondent pas au maximum d'une fonction (contrairement aux points du deuxième type). Pour ces deux types de landmarks, il faut que la correspondance dans les différents échantillons soit donnée. En effet, si l'on prend comme landmark le fond d'une dépression, il faut s'assurer que cette dépression apparaisse dans tous les échantillons et que ce fond soit bien défini pour tous les échantillons.

En plus de ces trois types de landmarks, il existe des *semi-landmarks* qui sont des points répartis le long d'une courbe ou d'une surface dont les extrémités sont définies par des landmarks. Ces semi-landmarks permettent de définir des contours. Il faut alors faire le choix du nombre de semi-landmarks qui seront utilisés pour représenter le contour. Ce nombre de semi-landmarks sera le même dans chaque échantillon pour assurer leur correspondance.

Il y a deux façons de relever les landmarks sur une structure anatomique. La première est de pointer directement les landmarks sur la structure, par exemple en utilisant un stylet 3D. Il faut alors

que l'expert puisse accéder manuellement aux emplacements des landmarks. La deuxième consiste à numériser les structures anatomiques grâce à des techniques d'imagerie utilisant par exemple des rayons X (CT-scan), la résonance magnétique nucléaire (IRM) ou des ultra-sons (échographies). On obtient alors une modélisation 3D de la structure anatomique sur laquelle les experts relèvent les landmarks à l'aide d'un logiciel de visualisation 3D.

1.2 Objectifs de la morphométrie

Lors d'une étude morphométrique d'objets, on étudie de façon quantitative les différences de formes entre ces objets. Une telle étude sert à répondre à deux questions : *quels sont les objets ayant des formes semblables et les objets ayant des formes différentes et en quoi les formes de deux objets sont semblables et en quoi ces formes sont différentes*. Nous allons définir ces questions de façon plus formelle.

Une *classification* de l'ensemble des échantillons est une partition. En mathématique, une *partition* d'un ensemble d'éléments $E = \{e_1, \dots, e_n\}$ est une séparation de E en plusieurs sous-ensembles P_1, \dots, P_k telle que tout élément e_i appartienne à un unique sous-ensemble P_j . Ainsi, l'union des P_i est égal à l'ensemble E et l'intersection entre deux sous-ensembles P_i et P_j est vide. Dans une classification, chaque sous-ensemble est appelée classe. On peut distinguer deux types de classifications.

classification
partition

Le premier type de classification est la classification de référence fournie par des experts. Les classifications de référence peuvent provenir d'informations différentes dont les degrés de certitude peuvent varier. Par exemple en anthropologie, une classification de référence peut provenir d'informations génétiques (dans ce cas cette classification est sûre), d'informations cliniques : est-ce qu'un patient présente telle ou telle pathologie (le diagnostic pouvant être erroné) ou d'informations visuelles (dans ce cas deux spécialistes peuvent ne pas fournir la même classification).

Le deuxième type de classification est la classification obtenue après avoir utilisé une méthode automatique d'analyse morphométrique. On quantifie les différences de forme entre les échantillons ce qui conduit à une classification. Cette classification peut être différente de la classification de référence, que ce soit par la composition des classes comme par le nombre de classes. En reprenant l'exemple de l'étude médicale, on peut imaginer qu'une pathologie peut ne pas avoir d'impact sur la forme ou avoir un impact proche une autre pathologie ou n'avoir un impact que dans certains cas. Dans ce cas, les classes que nous obtiendrons seront donc différentes des classes de références.

Lorsque l'on utilise une méthode de morphométrie, on souhaite non seulement déterminer les classes mais aussi connaître les propriétés des échantillons composant les classes. En effet, lorsque l'on souhaite comparer les formes d'échantillons, on cherche ce qui différencie les formes pour pouvoir décrire les classes obtenues.

caractérisation
d'une
classe

Dans cette thèse, on dira qu'une *propriété caractérise une classe* si cette propriété est vérifiée pour tous les échantillons de la classe. On dira qu'une *propriété discrimine une classe C_1 d'une classe C_2* si cette propriété est vérifiée pour tous les échantillons de la classe C_1 et que cette propriété n'est vérifiée par aucun échantillon de C_2 . Ainsi, une méthode de morphométrie doit nous permettre de trouver les propriétés qui discriminent les classes.

discrimination
de classes

1.3 Méthodes de morphométrie 3D à base de landmarks utilisées en anthropologie

Il existe de nombreux articles présentant un résumé des différentes méthodes de morphométries, que ce soit uniquement sur les méthodes basées sur des landmarks [1, 48, 68] ou sur toutes les méthodes [51]. Dans cette section, nous allons présenter les méthodes classiques en anthropologie de morphométrie basées sur des landmarks.

Nous allons illustrer ces méthodes sur trois configurations de cinq points en dimension deux. Ces configurations sont représentées par la Figure 1.2. La seule différence entre ces trois configurations réside dans la position du point 5.

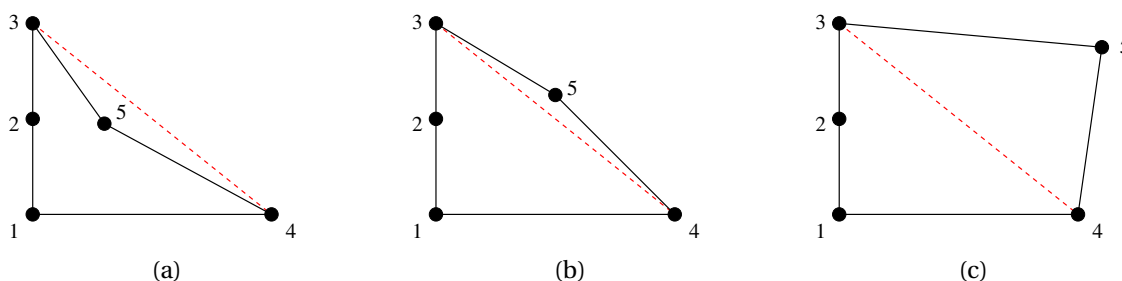


FIGURE 1.2 – Configurations de 5 points en 2 dimensions. Seul le point 5 a été déplacé.

A noter que beaucoup de méthodes de morphométrie utilisent un outil statistique appelé *Analyse en Composantes Principales* pour étudier les données. L'analyse en composantes principales est une méthode statistique qui fait partie du groupe des méthodes descriptives multidimensionnelles, c'est-à-dire qu'elle permet de traiter simultanément un grand nombre de variables numériques sans que l'on n'ait d'hypothèses probabilistes sur les variables (par exemple on ne suppose pas que ces variables sont distribuées selon des lois normales). L'analyse en composantes principales permet de représenter l'ensemble des échantillons à analyser dans un espace à deux ou trois dimensions, en minimisant la perte d'information.

On note p le nombre de variables. On dit que des observations sont centrées lorsque l'on a retranché à chaque observation la moyenne de la variable correspondante et que des observations sont réduites lorsque l'on a divisé chaque donnée par l'écart-type de la variable correspondante. L'inertie du nuage d'observations indique sa dispersion. Elle se calcule en faisant la somme des variances des variables. On cherche alors à projeter le nuage des observations centrées et réduites (qui est dans un espace de dimension p) sur un espace de dimension deux ou trois tout en maximisant l'inertie du nuage projeté. Il faut donc déterminer des vecteurs formant une base de l'espace projeté.

Le calcul des composantes principales est le suivant : Si les données sont centrées et réduites, on calcule la matrice de variance-covariance de l'ensemble des variables puis les valeurs propres et vecteurs propres de cette matrice. On trie ensuite les vecteurs propres selon les valeurs propres par ordre décroissant. Le principe est le même si les données sont uniquement centrées sauf qu'on

utilise comme matrice la matrice de corrélation de l'ensemble des variables. Dans cette méthode, les vecteurs propres sont appelés composantes principales.

Pour déterminer la qualité de l'analyse en composantes principales, on détermine pour chaque vecteur propre le pourcentage de la variance totale exprimée par le vecteur. Ce pourcentage est égal à la valeur propre divisée par la somme des p valeurs propres. Ainsi plus ce pourcentage est élevé pour un vecteur et plus les différences entre ces échantillons se font selon la direction de ce vecteur.

Une fois que l'on a trouvé les composantes principales, on représente les échantillons dans le plan formé par les deux premières composantes principales ou dans l'espace formé par les trois premières composantes principales. On en déduit une classification des échantillons en étudiant la répartition des échantillons dans cette représentation.

1.3.1 Les méthodes "traditionnelles"

Nous allons présenter les méthodes morphométriques utilisées dans les années 1960 et 1970. Pour plus de détails, nous renvoyons le lecteur à [45].

Les méthodes dites "traditionnelles" ne se basent pas directement sur les coordonnées tri-dimensionnelles des landmarks mais sur des mesures de longueurs, d'angles, de poids et d'aires. Elles consistent alors à réaliser une étude statistique de ces données, le plus souvent une analyse en composantes principales.

Nous allons étudier les configurations de points de la Figure 1.2 de cette façon. Comme seul le point 5 est modifié dans ces configurations, de nombreuses distances et angles sont identiques dans ces configurations. On peut par exemple étudier ces configurations de points en utilisant l'aire et l'angle $\widehat{453}$. La Figure 1.3 représente le graphique obtenu en utilisant l'aire en abscisse et l'angle $\widehat{453}$ en degré en ordonnée. On remarque que les configurations de points (a) et (b) sont proches l'une de l'autre et que la configuration de points (c) est plus éloignée. Ainsi, en utilisant l'aire et l'angle $\widehat{453}$ pour obtenir deux classes, la première classe serait composée des configurations (a) et (b) et la troisième classe serait composée de la configuration (c).

1.3.2 Méthodes par superposition

Lorsque l'on travaille à partir d'images en deux ou trois dimensions, il n'arrive que très rarement que les positions et les orientations des structures anatomiques à étudier soient les mêmes. De plus, lorsque les données n'ont pas toutes été acquises avec le même équipement, le repère dans lequel est modélisée la structure anatomique n'est pas forcément le même. Ces problèmes ont été l'une des motivations qui ont conduit à la création des méthodes de superposition.

Les méthodes de superposition ont pour but de superposer les différents échantillons, c'est-à-dire les mettre à la même échelle et de les aligner. Une fois que ces transformations ont été faites, les différences résiduelles de formes entre deux échantillons sont alors analysées.

Superposition de Procruste. Parmi toutes les méthodes de superposition, la plus utilisée est la méthode de superposition de Procruste [48].

Cette méthode utilise le *centroïde d'un échantillon* qui est défini comme étant le point dont centroïde

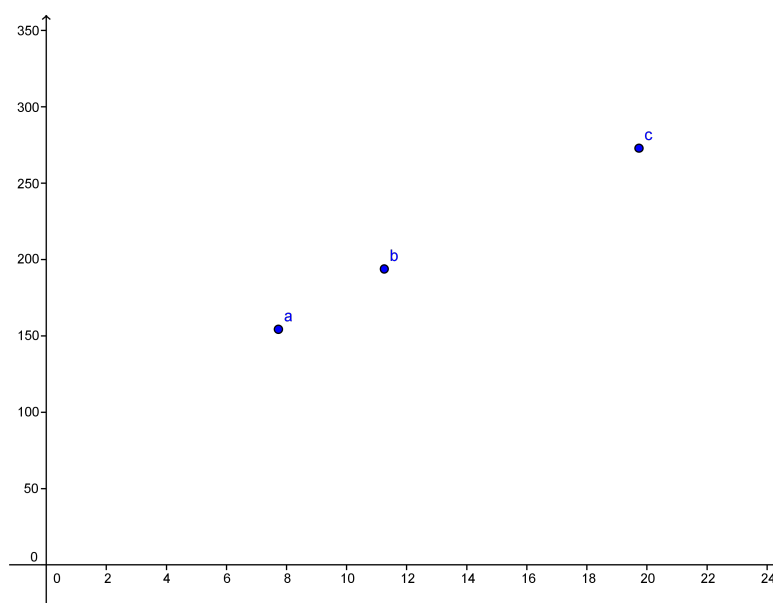


FIGURE 1.3 – Graphique représentant la répartition des trois configurations de points de la Figure 1.2 en utilisant l'aire en abscisse et l'angle $\widehat{453}$ en degré en ordonnée.

chaque coordonnée est la moyenne des coordonnées des landmarks de cette échantillon (en mathématiques, le centroïde s'appelle isobarycentre). On définit alors la *taille centroïdale d'un échantillon* comme étant égale à la somme des distances au carré entre le centroïde et chaque landmark de l'échantillon.

taille
centroïdale

Pour appliquer cette méthode, on commence par choisir un échantillon qui servira d'échantillon de référence et on calcule son centroïde. Pour chaque autre échantillon e , la méthode utilise trois étapes :

1. Translater l'échantillon e afin que son centroïde et le centroïde de l'échantillon de référence soient confondus.
2. Appliquer une homothétie à l'échantillon e afin que sa taille centroïdale soit la même que celle de l'échantillon de référence.
3. Faire une rotation 2D ou 3D de l'échantillon e afin de minimiser la somme des distances au carré entre les landmarks qui se correspondent dans l'échantillon e et l'échantillon de référence.

Il ne reste plus qu'à étudier les différences entre les échantillons superposés. Pour cela, nous allons calculer l'échantillon moyen. On note e^k le k -ème échantillon et \bar{e} l'échantillon moyen. Pour tout échantillon e , on note $e(i, j)$ la j -ème coordonnée du i -ème landmark. On a alors

$$\bar{e}(i, j) = \frac{\sum_{k=1}^m e^k(i, j)}{m}$$

où m est le nombre d'échantillons.

A partir de cet échantillon moyen, pour chaque échantillon e_k on calcule la matrice de résidus de e_k (noté R_k) qui est définie par $R_k(i, j) = \bar{e}(i, j) - e^k(i, j)$. Ainsi la matrice de résidus de chaque échantillon est une matrice de taille $n \times d$ où n est le nombre de landmarks et d est la dimension.

L'analyse de ces matrices de résidus nous permet d'obtenir les différences de formes entre les échantillons. La méthode la plus employée pour réaliser cette analyse consiste à faire une Analyse en Composantes Principales.

Exemple. Nous allons illustrer cette méthode en prenant comme exemple les trois configurations de la Figure 1.2 que nous allons analyser à l'aide du logiciel "Morphologika"¹ couramment utilisé en anthropologie (par exemple [52, 53, 62]).

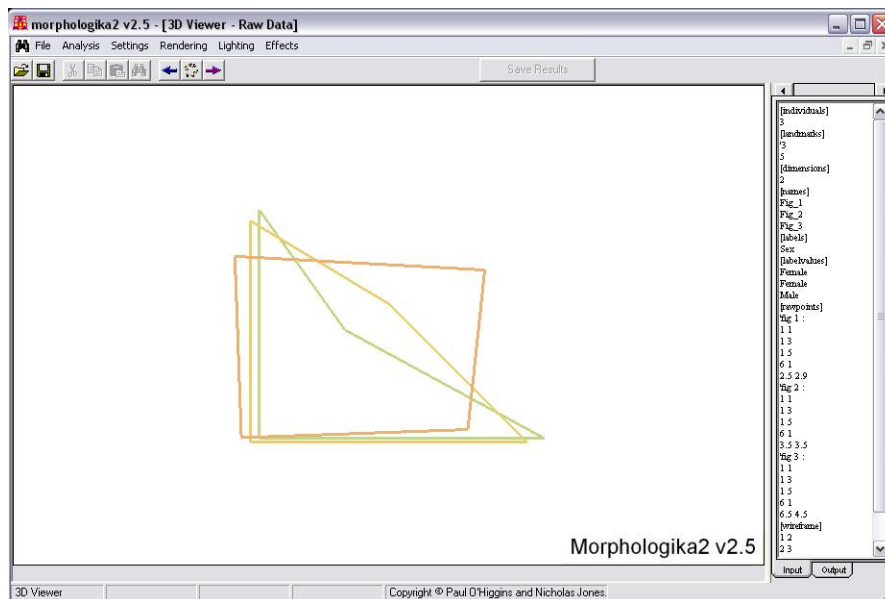


FIGURE 1.4 – Configurations de points de la Figure 1.2 après application de l'étape de superposition de Procruste. Ces figures ont été obtenues en utilisant le logiciel "Morphologika".

La Figure 1.4 représente le résultat de l'étape de superposition de Procruste, c'est-à-dire après translation, rotation et mise à l'échelle. On calcule alors les matrices de résidus des configurations de points après superposition, puis on réalise une Analyse en Composantes Principales en ne conservant que les deux premières composantes principales. Le résultat est présenté dans la Figure 1.5. Lorsque l'on calcule dans cet espace les distances entre les points a , b et c (correspondant respectivement aux configurations de points (a) , (b) et (c) après transformations), la distance entre a et b est d'environ 0,18 alors que la distance entre b et c est d'environ 0,40. Ainsi, si l'on cherche à obtenir deux classes à partir des configurations de points de la Figure 1.2, la première classe serait composée des configurations de points (a) et (b) et la deuxième classe serait composée de la configuration (c) .

1. Ce logiciel est disponible à l'adresse <https://sites.google.com/site/hymfme/resources> mais n'est plus maintenu ni développé. Il est remplacé par le logiciel "EVANT Toolbox" (<http://www.evan-society.org/>)

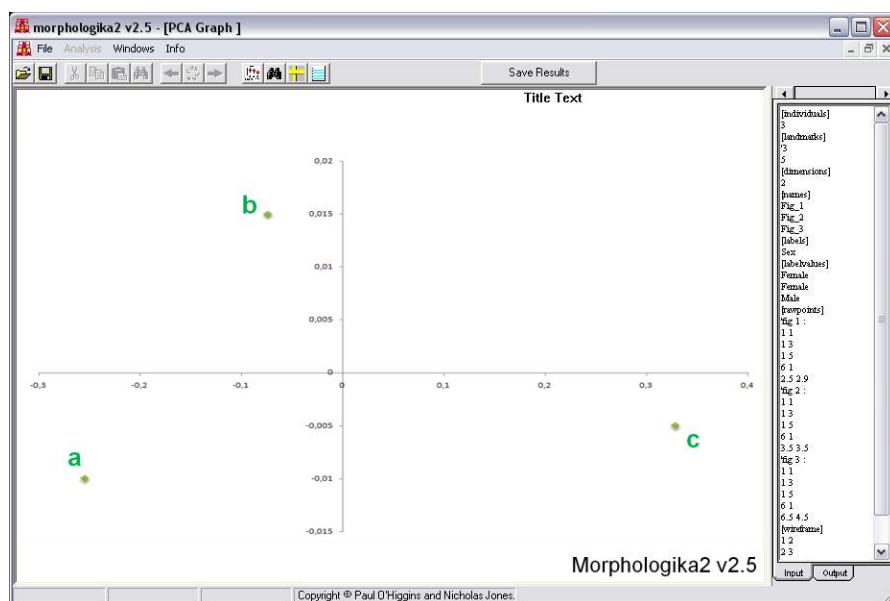


FIGURE 1.5 – Résultat de l'Analyse en Composante Principale en ne conservant que les deux premières composantes principales.

1.3.3 Méthodes par déformation

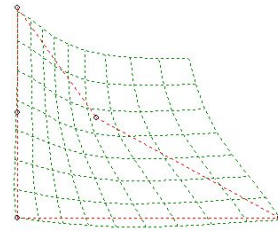
Le principe des méthodes de déformation est comparer deux échantillons en étudiant la déformation qu'il faut appliquer sur un échantillon pour qu'il corresponde exactement à l'autre échantillon. Lorsque l'on souhaite comparer plusieurs échantillons, on superpose d'abord les échantillons puis on calcule un échantillon moyen (voir Section 1.3.2) et chaque échantillon. On étudie alors les déformations entre l'échantillon moyen et les autres échantillons et leurs paramètres pour connaître les différences entre les formes.

L'une des méthodes de déformation les plus utilisées est basée sur des fonctions paramétriques de type "Thin-Plate Splines" [6]. L'idée de cette méthode provient du phénomène physique qui consiste à déformer une feuille mince de métal élastique pour lui donner une forme particulière. Cela permet d'extrapoler la déformation calculée sur les landmarks à tout l'espace et de la visualiser sous forme d'une grille déformée régulière. Les paramètres de la déformation permettent de calculer les modes de déformation principaux (Principal Warps).

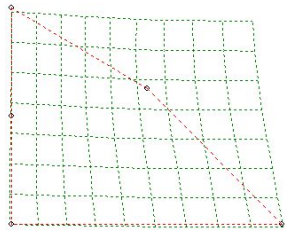
Exemple. La Figure 1.6 représente différentes déformations obtenues en appliquant cette méthode sur les configurations de points de la Figure 1.2. Les calculs des déformations ainsi que ces figures ont été obtenues en utilisant le logiciel "tpsSpln" disponible à l'adresse <http://life.bio.sunysb.edu/morph/>. La déformation de la grille permet de visualiser les différences entre les deux formes.



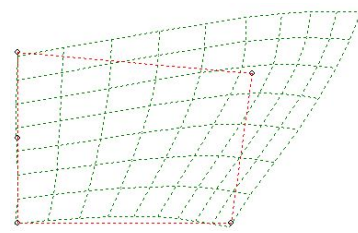
(a) Configuration de points qui est la configuration moyenne des configurations de la Figure 1.2.



(b) Déformation de la configuration de points moyenne vers celle de la Figure 1.2a.



(c) Déformation de la configuration de points moyenne vers celle de la Figure 1.2b.



(d) Déformation de la configuration de points moyenne vers celle de la Figure 1.2c.

FIGURE 1.6 – Déformations des configurations de points de la Figure 1.2. Ces figures ont été obtenues en utilisant le logiciel “tpsSpln” disponible à l’adresse <http://life.bio.sunysb.edu/morph/>.

1.3.4 Euclidean Distance Matrix Analysis

La méthode EDMA [40] (Euclidean Distance Matrix Analysis) a été conçue pour s’abstraire de l’étape de superposition, contrairement aux méthodes précédentes. Pour cela, cette méthode ne se base pas directement sur les coordonnées des landmarks mais sur la matrice des distances euclidiennes entre landmarks. Les rotations, translations et réflexions sont des isométries, c’est-à-dire qu’en appliquant une de ces transformations ou des combinaisons de ces transformations à un objet de l’espace, alors les distances sont conservées. Ainsi la matrice des distances euclidiennes inter-landmarks ne varie pas si l’on applique de telles transformations.

La matrice des distances euclidiennes d’un échantillon e est une matrice $n \times n$ où n est le nombre de landmarks de cet échantillon. En notant $FM(e)$ la matrice des distances et pour tout i et j de $\{1, \dots, n\}$, $FM_{i,j}(e)$ le coefficient de $FM(e)$ qui est à la ligne i et à la colonne j de la matrice, on a alors $FM_{i,j}(e) = d(l_i, l_j)$ où l_i et l_j sont les landmarks i et j de e et d est la distance euclidienne. Ainsi cette matrice est symétrique, tous les coefficients sont positifs ou nuls et les éléments diagonaux ($FM_{i,i}(e)$ pour tout $i \in \{1, \dots, n\}$) sont égaux à zéro. Les coefficients respectent également l’inégalité triangulaire c’est à dire que pour tout $i, j, k \in \{1, \dots, n\}$, $FM_{i,k}(e) + FM_{k,j}(e) \geq FM_{i,j}(e)$.

On a ainsi :

$$FM(e) = \begin{pmatrix} 0 & d(l_1, l_2) & d(l_1, l_3) & \cdots & d(l_1, l_n) \\ d(l_2, l_1) & 0 & d(l_2, l_3) & \cdots & d(l_2, l_n) \\ d(l_3, l_1) & d(l_3, l_2) & 0 & \cdots & d(l_3, l_n) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d(l_n, l_1) & d(l_n, l_2) & d(l_n, l_3) & \cdots & 0 \end{pmatrix}$$

Dans la méthode EDMA, pour comparer la forme d'échantillons, on compare les matrices des distances des échantillons. Ainsi on a $FM(e_1) = FM(e_2)$ si et seulement si les deux formes sont les mêmes. De même, il existe une constante c strictement positive telle que $FM(e_1) = c \times FM(e_2)$ si et seulement si on augmente la forme de e_2 d'un facteur c pour obtenir la forme de e_1 (pour $c \geq 1$) ou on réduit la forme de e_2 d'un facteur $1/c$ pour obtenir la forme de e_1 (pour $c \leq 1$).

Pour décrire la différence entre les formes de e_1 et de e_2 , on calcule la matrice $FDM(e_1, e_2)$ (pour "form difference matrix") de taille $n \times n$ où pour tout $i, j \in \{1, \dots, n\}$, $FDM_{i,j}(e_1, e_2) = FM_{i,j}(e_1)/FM_{i,j}(e_2)$ si $i \neq j$ et $FDM_{i,i}(e_1, e_2) = 1$. e_1 et e_2 ont la même forme si et seulement si tous les coefficients de $FDM(e_1, e_2)$ sont égaux à 1. De même, les formes de e_1 et de e_2 sont égales à un changement d'échelle près si et seulement si tous les coefficients de $FDM(e_1, e_2)$ qui ne sont pas sur la diagonale sont égaux à la même constante (cette constante est égale au coefficient d'agrandissement).

Pour pouvoir comparer plusieurs matrices FDM de façon statistiques, on calcule l'entier $t(e_1, e_2) = (\max FDM_{i,j}(e_1))/(\min FDM_{i,j}(e_2))$. Plus la valeur de $t(e_1, e_2)$ est grande et plus les formes sont différentes. Cette valeur à l'avantage d'être invariant au changement d'échelle, c'est à dire que $t(e_1, e_2) = 1$ si et seulement si il existe c strictement positif tel que $FM(e_1) = c \times FM(e_2)$.

Si l'on souhaite comparer les formes de deux ensembles d'échantillons $\{x_1, \dots, x_s\}$ et $\{y_1, \dots, y_t\}$, la méthode consiste à comparer la forme moyenne de $\{x_1, \dots, x_s\}$ et la forme moyenne $\{y_1, \dots, y_t\}$. Pour cela, on calcule la "form matrix" de chaque échantillon. Ensuite on calcule pour chaque ensemble la matrice de taille $n \times n$ dont les coefficients sont les moyennes des coefficients des "form matrices" des échantillons de cet ensemble. Par abus de notation, on note $FM(x)$ et $FM(y)$ ces matrices pour les ensembles $\{x_1, \dots, x_s\}$ et $\{y_1, \dots, y_t\}$. Formellement on a : $FM_{i,j}(x) = (\sum_{1 \leq e \leq s} FM_{i,j}(x_e))/s$. On utilise alors ces deux matrices comme s'il s'agissait de deux matrices calculées à partir de deux échantillons, c'est à dire qu'à partir de ces deux matrices, on calcule $FDM(x, y)$ et $t(x, y)$ pour quantifier les différences entre la forme moyenne de $\{x_1, \dots, x_s\}$ et la forme moyenne de $\{y_1, \dots, y_t\}$. On utilise alors un test de permutation sur la valeur de $t(x, y)$. Si le test de permutation vérifie l'hypothèse nulle, alors les deux ensembles d'échantillons $\{x_1, \dots, x_s\}$ et $\{y_1, \dots, y_t\}$ ne sont pas différents du point de vue de la forme.

Cette méthode peut également être utilisée pour calculer des modèles de croissance [69], c'est à dire pour tester si deux échantillons proviennent d'un même spécimen mais à des périodes différentes du développement (par exemple, si les échantillons E_1 et E_2 proviennent d'un enfant pris à des âges différents).

Exemple. Nous allons illustrer cette méthode en prenant comme exemple les trois configurations de la Figure 1.2. Nous obtenons alors les matrices de distances suivantes respectivement pour les configurations de points (a) , (b) et (c) :

$$FM(a) = \begin{pmatrix} 0 & 2 & 4 & 5 & 2,42 \\ 2 & 0 & 2 & 5,39 & 1,50 \\ 4 & 2 & 0 & 6,40 & 2,58 \\ 5 & 5,39 & 6,40 & 0 & 3,98 \\ 2,42 & 1,50 & 2,58 & 3,98 & 0 \end{pmatrix} \quad FM(b) = \begin{pmatrix} 0 & 2 & 4 & 5 & 3,54 \\ 2 & 0 & 2 & 5,39 & 2,55 \\ 4 & 2 & 0 & 6,40 & 2,92 \\ 5 & 5,39 & 6,40 & 0 & 3,54 \\ 3,54 & 2,55 & 2,92 & 3,54 & 0 \end{pmatrix}$$

$$FM(c) = \begin{pmatrix} 0 & 2 & 4 & 5 & 6,52 \\ 2 & 0 & 2 & 5,39 & 5,70 \\ 4 & 2 & 0 & 6,40 & 5,52 \\ 5 & 5,39 & 6,40 & 0 & 3,54 \\ 6,52 & 5,70 & 5,52 & 3,54 & 0 \end{pmatrix}$$

Nous avons calculé les “form difference matrix” pour tous les couples de configurations de points, nous avons calculé les valeurs de t pour tous les couples de configurations de points. On obtient alors $t(a, b) = 1,91$, $t(a, c) = 4,27$ et $t(b, c) = 2,24$. Ainsi en cherchant à obtenir deux classes à l’aide de la méthode EDMA, la première classe serait composée des configurations de points (a) et (b) et la deuxième classe serait composée de la configuration (c) .

Bien d’autres méthodes, notamment des méthodes sans landmarks, existent pour analyser de façon quantitative la forme de structures anatomiques, comme par exemple, en anatomie computationnelle (Computational Anatomy) [27, 57].

1.4 Limites de ces méthodes

Etudions maintenant les limites de ces différentes méthodes.

Les méthodes traditionnelles. Les méthodes traditionnelles ne se basent que sur un petit nombre de mesures (distances, angles, poids ou aires) et elles ne permettent pas de prendre en compte la forme dans sa globalité. Par exemple, en ne mesurant que la longueur et la largeur d’une forme en deux dimensions, il n’est pas possible de distinguer un losange, un rectangle et un ovale ayant la même largeur et la même longueur.

Les méthodes de superposition. Les méthodes de superposition sont souvent utilisées car elles produisent un graphique qui est facile à interpréter. Par contre ces méthodes sont sensibles au bruit puisque si l’on déplace légèrement un landmark alors le résultat de la superposition avec l’échantillon de référence ne sera pas le même. Un autre inconvénient de la méthode de superposition concerne l’étape de mise à l’échelle et de rotation des échantillons. En effet, comme le montre la Figure 1.4, la superposition de Procruste modifie les positions de tous les landmarks. Ainsi les méthodes de superposition peuvent transformer une déformation locale en une déformation globale (cet effet est appelé “Pinocchio effect” [82]). D’autres méthodes de superposition existent utilisant chacune un critère différent pour superposer les échantillons. Les résultats obtenus pour un jeu de données dépendent de la méthode (et donc du critère) utilisée [68].

Les méthodes de déformation. Les méthodes de déformations permettent de visualiser les différences entre deux formes, à l'aide de grilles régulières 2D ou 3D. Par contre ces méthodes ont plusieurs inconvénients. Le premier inconvénient concerne l'interprétation des résultats puisqu'il est très difficile de déterminer à quoi correspondent les paramètres des fonctions modélisant les déformations. De plus, ces méthodes sont sensibles au bruit puisqu'un léger déplacement d'un landmark modifie les paramètres de la fonction qui sert à modéliser la déformation pour passer d'un échantillon à un autre. Le second inconvénient est que les méthodes de déformation nécessitent de choisir un espace de fonctions paramétriques. De fait, les résultats peuvent dépendre plus du choix de la fonction à utiliser que des informations issues des landmarks [68]. Enfin, les méthodes de déformation partagent un inconvénient avec les méthodes de superposition puisque celles-ci utilisent également une étape d'alignement des échantillons.

La méthode Euclidean Distance Matrix Analysis. Contrairement aux méthodes précédentes, la méthode EDMA permet de comparer des échantillons sans réaliser d'alignements des échantillons. Par contre l'interprétation des résultats de cette méthode n'est pas aussi facile qu'avec les méthodes précédentes et ce pour deux raisons. La première raison vient du fait que le coefficient t servant à quantifier les différences entre les formes de deux échantillons n'est pas une distance (l'exemple de la Section 1.3.4 montre que cette valeur ne respecte pas l'inégalité triangulaire). On ne peut donc pas représenter les échantillons par des points dans un certain espace de sorte que la distance entre deux points soit égale à la valeur de t pour les deux échantillons correspondants. La deuxième raison provient du fait que la méthode est basée sur les distances et non sur les coordonnées des landmarks. En effet, cette méthode se basant sur les distances entre toutes les paires de landmarks, on obtient comme résultat quelles sont les distances qui sont modifiées entre les deux échantillons. Il est alors difficile de déterminer quels landmarks ont été déplacés entre deux échantillons et dans quels directions ont eu lieu ces déplacements [13].

Limites communes à ces méthodes. Toutes les méthodes que nous avons présentés dans ce chapitre permettent de déterminer des changements géométriques (différence de position de landmarks, de distances entre landmarks) entre les échantillons. Or en anthropologie, il arrive que des différences caractérisant les changements anatomiques ou évolutifs soient plus "structurels" que géométriques. Par exemple sur la Figure 1.7 il apparaît que la variation de l'angle basicranial est fondamentale pour caractériser les Hommes et les singes et est très important dans l'étude de l'évolution. Or les méthodes de morphologie géométrique ne mettent pas assez l'accent sur de telles différences puisque cette différence est liée à un faible déplacement de points.

En reprenant l'exemple de la Figure 1.2, on aimerait mettre en évidence la différence structurelle qu'il y a entre la configuration de points (a) et les configurations de points (b) et (c). En effet, dans la configuration (a) le point 5 est à l'intérieur du triangle 134 (donc le point 5 ne fait pas partie de l'enveloppe convexe de la configuration (a)), alors que le point 5 est à l'extérieur du triangle pour les deux autres configurations (donc le point 5 fait partie de l'enveloppe convexe des configurations (b) et (c)). Or lorsque l'on essayait d'obtenir deux classes que ce soit avec la superposition de Procruste (voir Section 1.3.2) ou la méthode EDMA (voir Section 1.3.4), la première classe était composée des configurations de points (a) et (b) et la deuxième classe était composée de la configuration (c). Ces

deux méthodes ne permettraient pas de classer les configurations de points de sorte qu'une classe contienne les configurations (b) et (c) car ces méthodes sont fondées sur des distances.

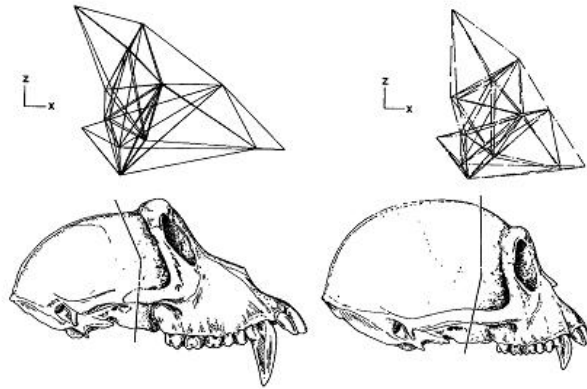


FIGURE 1.7 – Comparaison de deux crânes dont l'angle basicranial diffère (en bas). Les deux figures en haut représentent des landmarks situés au niveau de la face ou de la base du crâne.

Dans le reste de cette thèse nous allons donc proposer une méthode qui permet de s'affranchir des valeurs numériques tout en codant certaines propriétés "structurelles" de la forme.

Codage d'un modèle 3D par un matroïde orienté

Les matroïdes orientés [5, 20, 67] sont des outils mathématiques permettant entre autre de décrire les propriétés de configurations géométriques de façon combinatoire (c'est-à-dire de façon non numérique). La théorie des matroïdes orientés est développée depuis une quarantaine d'années dans le cadre de recherches théoriques en algèbre linéaire et en théorie des graphes.

Les matroïdes orientés ont été appliqués à de nombreux domaines théoriques tels que la théorie des graphes [5], la programmation linéaire [4], les arrangements de pseudo-lignes [17] et de pseudo-sphère [18], la pseudo-triangulation [61].

Par contre, la théorie des matroïdes orientés n'a été que peu appliquée en dehors des domaines mathématiques de la combinatoire et de la géométrie. Une de ces applications, peut-être la plus ancienne, concerne la classification des molécules chimiques à l'aide de matroïdes orientés [14], mais cette application reste très théorique. Des articles mentionnent régulièrement les matroïdes orientés dans des cadres plus applicatifs, mais soit se rapprochent des cadres théoriques habituels (par exemple [58] qui calcule des triangulations), soit s'éloignent complètement de la structure combinatoire (par exemple [74] qui code des positions relatives d'objets avec des signes inspirés des matroïdes orientés mais sans en être).

L'utilisation des matroïdes orientés permettrait de décrire les formes des structures anatomiques 2D ou 3D de façon purement combinatoire. Il n'y aurait donc plus de relations métriques entre les points ce qui permettrait notamment de ne plus avoir de problèmes de mise à l'échelle et de changement d'orientation.

Dans ce chapitre nous commençons par définir ce qu'est un matroïde orienté associé à un ensemble de points. Nous abordons ensuite les propriétés géométriques codées par un matroïde orienté associé à un ensemble de points puis nous finissons ce chapitre en introduisant une distance entre matroïdes orientés afin de pouvoir comparer les matroïdes orientés associés à des échantillons.

2.1 Comment associer un matroïde orienté à un ensemble de points 3D ?

Dans cette section nous commençons par donner une définition de ce qu'est un matroïde orienté associé à un ensemble de points. Nous donnons ensuite deux exemples (le premier en deux dimensions et le suivant en trois dimensions) de matroïdes orientés associés à des ensembles de points. Comme il existe différentes manières, toutes équivalentes, de définir un matroïde orienté, nous terminons cette section en donnant, à titre d'information, deux définitions équivalentes d'un matroïde orienté associé à un ensemble de points et nous abordons la notion de dualité qui est importante dans la théorie des matroïdes orientés. La dernière partie de cette section ne sera pas utilisée dans cette thèse.

2.1.1 Définition du matroïde orienté associé à un ensemble de points

La théorie des matroïdes orientés est une abstraction combinatoire de l'algèbre linéaire. En particulier les matroïdes orientés permettent de décrire et d'analyser les propriétés combinatoires de configurations géométriques. On peut appliquer la théorie des matroïdes orientés à des objets aussi différents que des configurations de points ou de vecteurs, des arrangements d'hyperplans ou de pseudosphères, ou même à des graphes. Il faut noter que la théorie des matroïdes orientés est très vaste et qu'il ne sera question ici que de matroïdes orientés associés aux configurations de points, appelés matroïdes orientés réalisables.

Définition d'un matroïde orienté

Un k -uplet d'un ensemble fini et ordonné \mathcal{E} est un sous-ensemble ordonné de \mathcal{E} de taille k . Dans tout ce qui suit, s'il n'y a pas d'ambiguïté, nous utiliserons l'abus de notation $abc\dots$ pour désigner le sous-ensemble ordonné $(a, b, c\dots)$ d'un ensemble fini \mathcal{E} .

Soit \mathcal{E} un ensemble de n points affines de \mathbb{R}^d . On dit que les points de \mathcal{E} sont en *position générale* si tout sous-espace réel de dimension $k < d$ ne contient pas plus que $k + 1$ points de \mathcal{E} . Ainsi dans \mathbb{R}^3 , un ensemble de points est en position générale s'il n'y a pas deux points confondus, trois points alignés et quatre points coplanaires. Si les points sont en position générale, on dit que le matroïde orienté est uniforme. Nous dirons alors que nous sommes dans le *cas uniforme* si les points de \mathcal{E} sont en position générale.

On associe à \mathcal{E} un ensemble \mathcal{V} de vecteurs de dimension $d + 1$. L'ensemble \mathcal{V} est obtenu à partir de l'ensemble \mathcal{E} de la manière suivante : pour tout point p de \mathcal{E} on associe le vecteur v de \mathcal{V} tel que la première coordonnée de v soit égale à 1 et que pour tout $i \in \{2, \dots, d + 1\}$, la i -ème coordonnée de v soit égale à la $(i - 1)$ -ème coordonnée de p . En notant p_i la i -ème coordonnée de p , le vecteur v

uplet

position
généralecas uni-
forme

associé à un point p de \mathcal{E} est alors l'image de p par la transformation f où

$$f : \mathbb{R}^d \rightarrow \mathbb{R}^{d+1}$$

$$p = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_d \end{pmatrix} \mapsto v = \begin{pmatrix} 1 \\ p_1 \\ p_2 \\ \vdots \\ p_d \end{pmatrix}$$

On a donc $\text{Card}(\mathcal{V}) = \text{Card}(\mathcal{E}) = n$.

Dans cette thèse, nous nous intéresserons plus particulièrement aux cas de points en trois dimensions, les vecteurs appartiendront donc à \mathbb{R}^4 .

Pour illustrer la théorie des matroïdes orientés, nous allons nous baser sur un exemple de 5 points en trois dimensions donnée par la Figure 2.1.

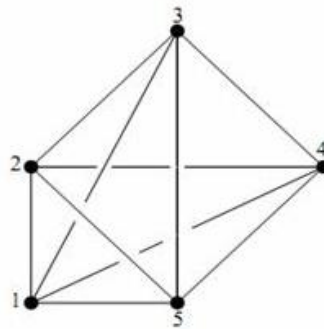


FIGURE 2.1 – Exemple de 5 points en 3 dimensions

La matrice suivante représente les coordonnées de ces points dans la base canonique.

$$\begin{pmatrix} 1 & 1 & 2 & 3 & 2 \\ 1 & 2 & 3 & 2 & 1 \\ 2 & 2 & 2 & -1 & 3 \end{pmatrix}$$

Ainsi l'ensemble \mathcal{V} se calcule en appliquant la transformation f à la matrice précédente. On obtient alors

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 3 & 2 \\ 1 & 2 & 3 & 2 & 1 \\ 2 & 2 & 2 & -1 & 3 \end{pmatrix}$$

Le matroïde orienté associé à l'ensemble de points \mathcal{E} est en réalité calculé à partir de l'ensemble de vecteurs \mathcal{V} .

L'ensemble des *bases d'un matroïde orienté* est un ensemble de parties de \mathcal{V} , chaque base or-

bases d'un
matroïde
orienté

donnée étant munie d'un signe appelé chirotope et noté χ . Une partie de \mathcal{V} est une base si c'est un ensemble linéairement indépendant de cardinal maximal. Les bases ont donc toutes pour cardinal $d + 1$ et on appelle cette valeur *rang du matroïde orienté*. Pour calculer les chirotopes des bases, on commence par écrire tous les vecteurs de \mathcal{V} dans une base de référence. Le chirotope d'une base ordonnée est le signe du déterminant de la matrice M où M est la matrice dont les colonnes sont les éléments de cette base écrite sur la base de référence.

Si un sous-ensemble de \mathcal{V} de cardinal $d + 1$ n'est pas linéairement indépendant, alors cette ensemble n'est pas une base et le déterminant sera donc nul. On peut alors étendre la définition de chirotope en incluant le cas où le déterminant est nul.

$$\chi : \begin{array}{ccc} \mathcal{V}^{d+1} & \longrightarrow & \{-, 0, +\} \\ (v_{i_1}, \dots, v_{i_{d+1}}) & \longmapsto & \text{sign}(\det(v_{i_1}, \dots, v_{i_{d+1}})) \end{array}$$

Comme les colonnes de la matrice sont écrites sur la base de référence, si le chirotope d'une base ordonnée est positif cela signifie que l'orientation de cette base est la même que l'orientation de la base de référence.

On peut remplacer la fonction χ par son opposée, ce qui correspond à prendre comme base de référence une base qui a l'orientation inverse.

Par abus de langage, on dira qu'un $(d + 1)$ -uplet de \mathcal{E} est une base si l'ensemble des vecteurs associés à ce $(d + 1)$ -uplet est une base. On parlera également du chirotope d'un $(d + 1)$ -uplet de \mathcal{E} pour évoquer le chirotope de l'ensemble des vecteurs associé à ce $(d + 1)$ -uplet de points.

On rappelle qu'un *hyperplan de \mathbb{R}^d* est un sous-espace de dimension $d - 1$.

Si un $(d + 1)$ -uplet de points de \mathcal{E} est une base, cela signifie que ce $(d + 1)$ -uplet forme un simplexe. Un *simplexe de \mathbb{R}^d* est l'enveloppe convexe de $(d + 1)$ points n'appartenant pas à un même hyperplan. Un simplexe de \mathbb{R}^2 est un triangle et un simplexe de \mathbb{R}^3 est un tétraèdre. La notion de simplexe est donc une généralisation du triangle et du tétraèdre en dimension quelconque.

Ainsi, le chirotope d'un $(d + 1)$ -uplet de points de \mathcal{E} est nul si et seulement si ces $d + 1$ points appartiennent à un hyperplan de \mathbb{R}^d . Si ces $d + 1$ points appartiennent à un hyperplan de \mathbb{R}^d alors le chirotope de ce $(d + 1)$ -uplet de points de \mathcal{E} correspond à l'orientation du simplexe formé par ces points. En effet, comme il n'existe que deux orientations possibles pour les simplexes, toutes les bases qui auront le même signe indiqueront les simplexes ayant la même orientation. La Figure 2.2 représente les deux orientations possibles d'un tétraèdre ABCD. On remarque alors que pour passer d'une orientation à l'autre, on peut par exemple déplacer le point A pour qu'il traverse le plan formé par les autres points. Cette remarque est vraie en dimension quelconque.

Dans le cas où les points sont en position générale, tous les sous-ensemble de \mathcal{V} de cardinal $d + 1$ sont donc linéairement indépendants. Ainsi tous les $(d + 1)$ -uplets de \mathcal{E} sont des bases et forment des simplexes.

D'après ce que nous venons de voir, le chirotope d'un matroïde orienté M (noté χ_M) est une application qui associe à chaque $(d + 1)$ -uplet de vecteurs de \mathcal{V} (ou de $(d + 1)$ -uplet de points de \mathcal{E} par abus de langage) un signe de $\{-, 0, +\}$. De même, le chirotope d'un $(d + 1)$ -uplet B dans un matroïde orienté M (noté $\chi_M(B)$) est un signe de $\{-, 0, +\}$. Dans la suite, nous serons amenés à travailler avec un ensemble de matroïdes orientés $\{M_1, \dots, M_m\}$ où m est le nombre d'échantillons que l'on souhaite étudier. Par souci de simplicité on parlera du signe d'un $(d + 1)$ -uplet (respectivement d'une

rang

hyperplan
de \mathbb{R}^d
simplexeorientation
d'un
simplexe

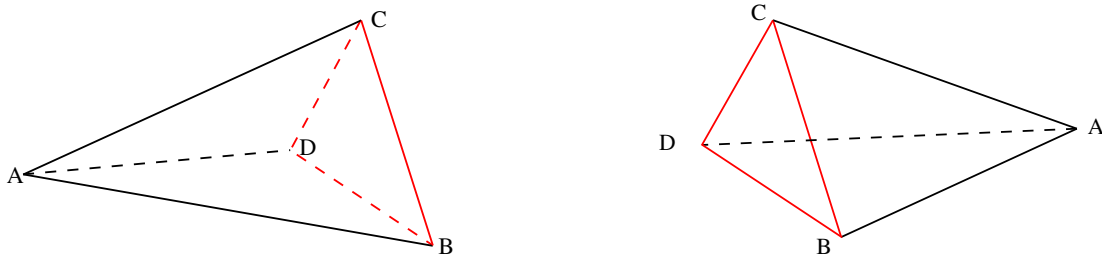


FIGURE 2.2 – Les deux orientations possibles pour le tétraèdre ABCD

base) dans le matroïde orienté M_i pour désigner le chirotope de ce $(d+1)$ -uplet (respectivement de dette base) dans le matroïde orienté M_j . Ainsi nous essayerons de réserver le terme de chirotope pour désigner l'application.

Le chirotope d'un matroïde orienté est une application *alternée* puisque le déterminant de matrices est une application alternée. Cela signifie que pour toute permutation π de $\{1, \dots, d+1\}$ et pour tout $(d+1)$ -uplet de vecteurs $(v_{i_1}, \dots, v_{i_{d+1}})$, on a

alterné

$$\chi(v_{i_1}, \dots, v_{i_{d+1}}) = (-1)^{\text{sgn}(\pi)} \chi(v_{i_{\pi(1)}}, \dots, v_{i_{\pi(d+1)}})$$

où $\text{sgn}(\pi)$ est la signature de la permutation π .

On va représenter le chirotope d'un matroïde orienté (c'est-à-dire l'application χ) par un vecteur de $\{-, 0, +\}$. Comme le chirotope est une application alternée, il n'est pas nécessaire de conserver les signes pour tous les $(d+1)$ -uplet de \mathcal{E} . Il suffit en fait de conserver un signe par classe de permutation. Pour cela, nous avons choisi d'ordonner les éléments de \mathcal{E} et de conserver le signe du plus petit $(d+1)$ -uplet dans l'ordre lexicographique pour chaque classe de permutation. En d'autres termes, on conserve un seul signe pour chaque sous-ensemble de \mathcal{E} de taille $d+1$. Nous représentons alors le chirotope d'un matroïde orienté par un vecteur de $\{-, 0, +\}$ de taille $\binom{n}{d+1}$. Dans la suite de cette thèse, on note $\mathcal{B}(\mathcal{E})$ l'ensemble des plus petits $(d+1)$ -uplets de \mathcal{E} dans l'ordre lexicographique pour chaque classe de permutation. Ainsi pour chaque $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$, on a $B = (e_1, \dots, e_{d+1})$ avec pour tout $i \in \{1, \dots, d+1\}$, $e_i \in \mathcal{E}$ et pour tout $i, j \in \{1, \dots, d+1\}$, $e_i < e_j$.

$\mathcal{B}(\mathcal{E})$

Soit χ le chirotope d'un matroïde orienté. On définit le *vecteur représentatif du chirotope* χ , le vecteur de $\{-, 0, +\}$ de taille $\binom{n}{d+1}$ dont le i -ème élément est égal au signe dans χ du i -ème $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ dans l'ordre lexicographique.

Dans la suite de cette thèse, nous travaillerons sur des vecteurs représentatifs de chirotopes que ce soit par exemple pour coder en machine des chirotopes ou pour comparer des configurations de points entre elles. On se permettra par commodité d'employer le terme de *chirotope* pour désigner le vecteur représentatif d'un chirotope.

Revenons sur l'exemple de la Figure 2.1. Nous avons donc $\binom{5}{4} = 5$ quadruplets de points. Comme les points sont en position générale (il n'y a ni deux points confondus, ni trois points alignés, ni quatre points coplanaires), les chirotopes seront donc non nuls. Le calcul des bases nous donne donc le résultat suivant :

$$\begin{array}{lll} \chi(1\ 2\ 3\ 4) = + & \chi(1\ 2\ 3\ 5) = - & \chi(1\ 2\ 4\ 5) = - \\ \chi(1\ 3\ 4\ 5) = - & & \chi(2\ 3\ 4\ 5) = - \end{array}$$

Il y a donc cinq bases dont seule la première a la même orientation que la base canonique (puisque les vecteurs sont écrits dans la base canonique, la base canonique sert de base de référence).

2.1.2 Exemples de matroïdes orientés associés à des configurations de points en deux dimensions

Nous allons nous servir de la Figure 1.2 qui nous a servi d'exemple dans le chapitre précédent, pour illustrer les notions que nous venons de voir.

Les matroïdes orientés ne codent que la position relative des points les uns par rapport aux autres. Or dans les trois configurations de points, seule la position du point 5 a été modifiée. Ce point est à l'intérieur du triangle 1 3 4 pour la configuration (a) et à l'extérieur du triangle pour les configurations (b) et (c). Ainsi les matroïdes orientés associés aux configurations de points (b) et (c) seront les mêmes.

On peut calculer le chirotope des matroïdes orientés associé aux configurations (a), (b) et (c). En les notant respectivement χ_a , χ_b et χ_c , on obtient alors les signes suivants :

$\chi_a(1\ 2\ 3) = 0$	$\chi_b(1\ 2\ 3) = 0$	$\chi_c(1\ 2\ 3) = 0$
$\chi_a(1\ 2\ 4) = -$	$\chi_b(1\ 2\ 4) = -$	$\chi_c(1\ 2\ 4) = -$
$\chi_a(1\ 2\ 5) = -$	$\chi_b(1\ 2\ 5) = -$	$\chi_c(1\ 2\ 5) = -$
$\chi_a(1\ 3\ 4) = -$	$\chi_b(1\ 3\ 4) = -$	$\chi_c(1\ 3\ 4) = -$
$\chi_a(1\ 3\ 5) = -$	$\chi_b(1\ 3\ 5) = -$	$\chi_c(1\ 3\ 5) = -$
$\chi_a(1\ 4\ 5) = +$	$\chi_b(1\ 4\ 5) = +$	$\chi_c(1\ 4\ 5) = +$
$\chi_a(2\ 3\ 4) = -$	$\chi_b(2\ 3\ 4) = -$	$\chi_c(2\ 3\ 4) = -$
$\chi_a(2\ 3\ 5) = -$	$\chi_b(2\ 3\ 5) = -$	$\chi_c(2\ 3\ 5) = -$
$\chi_a(2\ 4\ 5) = +$	$\chi_b(2\ 4\ 5) = +$	$\chi_c(2\ 4\ 5) = +$
$\chi_a(3\ 4\ 5) = -$	$\chi_b(3\ 4\ 5) = +$	$\chi_c(3\ 4\ 5) = +$

Comme les points 1 2 3 sont alignés le triplet 1 2 3 n'est pas une base et le signe de ce triplet est alors 0. On remarque que la seule différence concerne l'orientation de la base formée par les points 3, 4, 5. Cela signifie que la base 3 4 5 a la même orientation que la base de référence pour les configurations (b) et (c) et a l'orientation opposée pour la configuration (a).

2.1.3 Exemples de matroïdes orientés associés à des configurations de points en trois dimensions

Nous allons maintenant illustrer les notions vues précédemment par un exemple en trois dimensions. Considérons les configurations de points données dans la Figure 2.3.

Ces représentations correspondent à des points en trois dimensions dont les coordonnées sont respectivement :

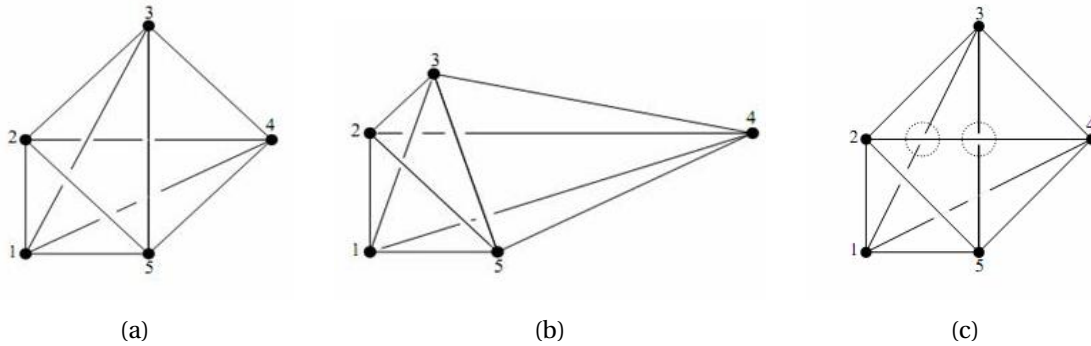


FIGURE 2.3 – Configurations de 5 points en 3 dimensions

$$\begin{pmatrix} 1 & 1 & 2 & 3 & 2 \\ 1 & 2 & 3 & 2 & 1 \\ 2 & 2 & 2 & -1 & 3 \end{pmatrix} \begin{pmatrix} 1 & 1 & 2 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \\ 2 & 2 & 2 & -1 & 3 \end{pmatrix} \begin{pmatrix} 1 & 1 & 2 & 3 & 2 \\ 1 & 2 & 3 & 2 & 1 \\ 2 & 2 & 2 & 3.25 & 3 \end{pmatrix}$$

Comme les matroïdes orientés ne traduisent que les positions relatives de points les uns par rapport aux autres, les matroïdes orientés associés aux configurations (a) et (b) seront les mêmes. En effet, en passant de la configuration (a) à la configuration (b), seul le point 4 a été déplacé mais sans qu'il ne traverse de plan formé par les autres points, ou en d'autres termes sans qu'un tétraèdre ordonné ne change d'orientation. Par contre, en passant de l'une des premières configurations à la configuration (c), il y a bien eu une modification au niveau de la position relative des points. La droite (2 4) est en dessous des droites (1 3) et (3 5) dans les Figures 2.3a et 2.3b, mais est au dessus de ces mêmes droites dans la Figure 2.3c.

On peut alors calculer le chirotope des matroïdes orientés associé aux configurations (a), (b) et (c). En les notant respectivement χ_a , χ_b et χ_c , on obtient alors les signes suivants :

$$\begin{array}{lll} \chi_a(1\ 2\ 3\ 4) = - & \chi_b(1\ 2\ 3\ 4) = - & \chi_c(1\ 2\ 3\ 4) = + \\ \chi_a(1\ 2\ 3\ 5) = + & \chi_b(1\ 2\ 3\ 5) = + & \chi_c(1\ 2\ 3\ 5) = + \\ \chi_a(1\ 2\ 4\ 5) = + & \chi_b(1\ 2\ 4\ 5) = + & \chi_c(1\ 2\ 4\ 5) = + \\ \chi_a(1\ 3\ 4\ 5) = + & \chi_b(1\ 3\ 4\ 5) = + & \chi_c(1\ 3\ 4\ 5) = + \\ \chi_a(2\ 3\ 4\ 5) = + & \chi_b(2\ 3\ 4\ 5) = + & \chi_c(2\ 3\ 4\ 5) = - \end{array}$$

Ainsi on remarque par exemple qu'il y a une différence dans l'orientation de la base formée par les points 1, 2, 3 et 4 entre la configuration (a) et la configuration (c). Cela signifie que la position relative de ces points n'est pas la même, c'est-à-dire par exemple que le point 4 n'est pas du même côté du triangle formé par les points 1, 2 et 3.

2.1.4 Quelques définitions équivalentes du matroïde orientés associé à un ensemble de points

La définition d'un matroïde orienté n'est pas unique, il existe différentes manières de définir un matroïde orienté à partir d'axiomes. Nous donnons ici deux définitions du matroïde orientés associé à un ensemble de points, utilisant l'ensemble des cocircuits pour la première et l'ensemble des circuits pour la deuxième. Ces définitions sont données à titre d'information mais ne sont pas utilisées dans cette thèse.

Les cocircuits. L'ensemble des *cocircuits* est un ensemble de parties signées de V . Une partie de \mathcal{V} est un *hyperplan du matroïde orienté* si elle engendre un hyperplan de \mathbb{R}^{d+1} (sous-espace de dimension d) et si elle est égale à l'intersection de \mathcal{V} avec cet hyperplan. Les supports des cocircuits sont les complémentaires des hyperplans dans \mathcal{V} . Un hyperplan sépare l'espace en deux demi-espaces, un demi-espace est choisi comme étant positif et l'autre comme étant négatif. Chaque élément du support du cocircuit a donc un signe qui est le signe du demi-espace auquel il appartient. On notera un cocircuit par une liste n signes (un par vecteur). Le i -ème signe de la liste est + (respectivement -) si le vecteur v_i appartient au demi-espace positif (resp. négatif) et le i -ème signe de la liste est 0 si le vecteur v_i appartient à l'hyperplan.

Bien sûr, si C^* est un cocircuit, alors $-C^*$ est aussi un cocircuit.

Si les points sont en position générale, chaque d -uplet de \mathcal{V} est un hyperplan. Il y a donc exactement $2 \cdot \binom{n}{d}$ cocircuits (le facteur 2 venant du fait que C^* et $-C^*$ sont des cocircuits). On se contente alors de calculer la moitié des cocircuits (soit $\binom{n}{d}$ cocircuits) et l'on obtient la totalité des cocircuits en prenant l'opposé de chaque cocircuit calculé. Par contre si les points ne sont pas en position générale deux phénomènes font qu'il y ait moins de cocircuits. Le premier phénomène est qu'il peut exister au moins un d -uplet de \mathcal{V} qui ne forme pas un hyperplan (s'ils génèrent un sous-espace de dimension strictement inférieur à d). Le deuxième phénomène est qu'un même hyperplan peut être formé de plus de d éléments de \mathcal{V} .

Dans le cas de points en deux dimensions, les vecteurs sont de dimension trois. Les hyperplans seront donc des sous-espaces de \mathbb{R}^3 dimension deux, donc des plans de \mathbb{R}^3 . Ainsi un hyperplan du matroïde orienté est l'ensemble des vecteurs de \mathcal{V} appartenant au plan défini par deux vecteurs quelconques de \mathcal{V} non colinéaires. Or les points de \mathcal{E} appartiennent à un même plan de \mathbb{R}^3 (le plan d'équation $z = 1$). L'intersection de l'hyperplan avec ce plan forme une droite (passant par les deux points correspondant aux deux vecteurs définissant l'hyperplan). Le signe d'un élément du cocircuit indique dans quel demi-espace se trouve le vecteur par rapport à l'hyperplan ; ce qui revient donc à dire dans quelle partie du plan se trouve le point par rapport à la droite. Ainsi les cocircuits indiquent par des + les points qui sont d'un côté de la droite et par des - les points qui sont de l'autre côté de la droite (les points appartenant à la droite sont notés par un 0).

De même, dans le cas de points en 3 dimensions, les cocircuits indiqueront la position des points par rapport à des plans. Ces plans seront définis par trois points quelconques de \mathcal{E} non alignés.

Illustrons la notion de cocircuits sur l'exemple de la Figure 2.1. Comme les points sont en position générale, tous les triplets de vecteurs définissent un hyperplan. Il y a alors $\binom{5}{3} = 10$ hyperplans

donc $2 \cdot \binom{5}{3} = 20$ cocircuits. Par exemple, les cocircuits $0\ 0\ 0\ +\ -$ et $0\ 0\ 0\ -\ +$ sont opposés et correspondent tous les deux au plan formé par les points 1 2 et 3. Dans le premier cocircuit, le demi-espace positif est celui dans lequel est situé le point 4 alors que dans le deuxième cocircuit le point 4 est dans le demi-espace négatif. On notera alors $C^*(1\ 2\ 3)$ l'un de ces deux cocircuits pour indiquer qu'il correspond au plan formé par les points 1 2 et 3. En calculant la moitié des cocircuits nous obtenons le résultat suivant (l'autre moitié est obtenue en changeant tous les signes dans chacun des cocircuits) :

$$\begin{aligned}
 C^*(1\ 2\ 3) &= 0\ 0\ 0\ +\ - \\
 C^*(1\ 2\ 4) &= 0\ 0\ -\ 0\ - \\
 C^*(1\ 2\ 5) &= 0\ 0\ +\ +\ 0 \\
 C^*(1\ 3\ 4) &= 0\ +\ 0\ 0\ - \\
 C^*(1\ 3\ 5) &= 0\ -\ 0\ +\ 0 \\
 C^*(1\ 4\ 5) &= 0\ -\ -\ 0\ 0 \\
 C^*(2\ 3\ 4) &= -\ 0\ 0\ 0\ - \\
 C^*(2\ 3\ 5) &= +\ 0\ 0\ +\ 0 \\
 C^*(2\ 4\ 5) &= +\ 0\ -\ 0\ 0 \\
 C^*(3\ 4\ 5) &= +\ +\ 0\ 0\ 0
 \end{aligned}$$

Nous allons maintenant calculer les cocircuits des configurations de points en deux dimensions de la Figure 1.2. Le calcul de la moitié des cocircuits nous donne le résultat suivant (l'autre moitié est obtenue en changeant tous les signes dans chacun des cocircuits)

$C_a^*(1\ 2\ 3) = 0\ 0\ 0\ -\ -$	$C_b^*(1\ 2\ 3) = 0\ 0\ 0\ -\ -$	$C_c^*(1\ 2\ 3) = 0\ 0\ 0\ -\ -$
$C_a^*(1\ 4) = 0\ +\ +\ 0\ +$	$C_b^*(1\ 4) = 0\ +\ +\ 0\ +$	$C_c^*(1\ 4) = 0\ +\ +\ 0\ +$
$C_a^*(1\ 5) = 0\ +\ +\ -\ 0$	$C_b^*(1\ 5) = 0\ +\ +\ -\ 0$	$C_c^*(1\ 5) = 0\ +\ +\ -\ 0$
$C_a^*(2\ 4) = -\ 0\ +\ 0\ +$	$C_b^*(2\ 4) = -\ 0\ +\ 0\ +$	$C_c^*(2\ 4) = -\ 0\ +\ 0\ +$
$C_a^*(2\ 5) = -\ 0\ +\ -\ 0$	$C_b^*(2\ 5) = -\ 0\ +\ -\ 0$	$C_c^*(2\ 5) = -\ 0\ +\ -\ 0$
$C_a^*(3\ 4) = -\ -\ 0\ 0\ -$	$C_b^*(3\ 4) = -\ -\ 0\ 0\ +$	$C_c^*(3\ 4) = -\ -\ 0\ 0\ +$
$C_a^*(3\ 5) = -\ -\ 0\ +\ 0$	$C_b^*(3\ 5) = -\ -\ 0\ -\ 0$	$C_c^*(3\ 5) = -\ -\ 0\ -\ 0$
$C_a^*(4\ 5) = +\ +\ -\ 0\ 0$	$C_b^*(4\ 5) = +\ +\ +\ 0\ 0$	$C_c^*(4\ 5) = +\ +\ +\ 0\ 0$

Le premier cocircuit indique que les points 1, 2 et 3 sont alignés et que les points 4 et 5 sont du même coté de cette droite. Les différences entre les cocircuits du matroïdes orientés associé à la configuration (a) et les cocircuits des matroïdes orientés associés aux configurations (b) et (c) se situent sur les trois derniers cocircuits. Par exemple le dernier cocircuit indique que si l'on considère la droite formé par les points 4 et 5, le point 3 est du même coté de la droite que les points 1 et 2 pour les configurations (b) et (c) et qu'il est de l'autre coté de la droite pour la configuration (a).

De même, le calcul de la moitié des cocircuits des configurations de points de la Figure 2.3 nous donne le résultat suivant (l'autre moitié est obtenue en changeant tous les signes dans chacun des cocircuits)

$$\begin{array}{lll}
C_a^*(1\ 2\ 3) = 0\ 0\ 0\ +\ - & C_b^*(1\ 2\ 3) = 0\ 0\ 0\ +\ - & C_c^*(1\ 2\ 3) = 0\ 0\ 0\ -\ - \\
C_a^*(1\ 2\ 4) = 0\ 0\ -\ 0\ - & C_b^*(1\ 2\ 4) = 0\ 0\ -\ 0\ - & C_c^*(1\ 2\ 4) = 0\ 0\ +\ 0\ - \\
C_a^*(1\ 2\ 5) = 0\ 0\ +\ +\ 0 & C_b^*(1\ 2\ 5) = 0\ 0\ +\ +\ 0 & C_c^*(1\ 2\ 5) = 0\ 0\ +\ +\ 0 \\
C_a^*(1\ 3\ 4) = 0\ +\ 0\ 0\ - & C_b^*(1\ 3\ 4) = 0\ +\ 0\ 0\ - & C_c^*(1\ 3\ 4) = 0\ -\ 0\ 0\ - \\
C_a^*(1\ 3\ 5) = 0\ -\ 0\ +\ 0 & C_b^*(1\ 3\ 5) = 0\ -\ 0\ +\ 0 & C_c^*(1\ 3\ 5) = 0\ -\ 0\ +\ 0 \\
C_a^*(1\ 4\ 5) = 0\ -\ -\ 0\ 0 & C_b^*(1\ 4\ 5) = 0\ -\ -\ 0\ 0 & C_c^*(1\ 4\ 5) = 0\ -\ -\ 0\ 0 \\
C_a^*(2\ 3\ 4) = -\ 0\ 0\ 0\ - & C_b^*(2\ 3\ 4) = -\ 0\ 0\ 0\ - & C_c^*(2\ 3\ 4) = +\ 0\ 0\ 0\ + \\
C_a^*(2\ 3\ 5) = +\ 0\ 0\ +\ 0 & C_b^*(2\ 3\ 5) = +\ 0\ 0\ +\ 0 & C_c^*(2\ 3\ 5) = +\ 0\ 0\ -\ 0 \\
C_a^*(2\ 4\ 5) = +\ 0\ -\ 0\ 0 & C_b^*(2\ 4\ 5) = +\ 0\ -\ 0\ 0 & C_c^*(2\ 4\ 5) = +\ 0\ +\ 0\ 0 \\
C_a^*(3\ 4\ 5) = +\ +\ 0\ 0\ 0 & C_b^*(3\ 4\ 5) = +\ +\ 0\ 0\ 0 & C_c^*(3\ 4\ 5) = +\ -\ 0\ 0\ 0
\end{array}$$

En prenant par exemple le cocircuit correspondant aux points 3, 4 et 5, on remarque que les signes des points 1 et 2 sont les mêmes dans la configuration (a) alors qu'ils sont différents dans la configuration (c). Cette différence vient du fait que les points 1 et 2 sont du même côté du plan formé par les points 3, 4 et 5 dans la configuration (a) alors qu'ils sont chacun d'un côté différent dans la configuration (c).

Les circuits. L'ensemble des *circuits* est un ensemble de parties signées de \mathcal{V} . Les supports des circuits sont les parties linéairement dépendantes minimales pour l'inclusion. Chaque élément du support du circuit a donc un signe qui est le signe du coefficient dans une relation de dépendance linéaire des éléments de son support \underline{C} . Formellement on a :

Soit $\underline{C} = (v_{i_1}, \dots, v_{i_k})$ un support d'un circuit C . Alors il existe $\lambda_1, \dots, \lambda_k \in \mathbb{R}$ tels que $\sum_j \lambda_j v_{i_j} = 0$. On a alors $C = (C^+, C^-)$ avec $C^+ = \{v_{i_j} \mid \lambda_j > 0\}$ et $C^- = \{v_{i_j} \mid \lambda_j < 0\}$.

De même que pour les cocircuits, si C est un circuit, alors $-C$ est aussi un circuit. Dans le cas de points en position générale, nous avons vu que tout $(d+1)$ -uplet de \mathcal{V} forme une base. Ces $(d+1)$ vecteurs sont donc linéairement indépendants. Par contre en rajoutant un vecteur, ces $(d+2)$ vecteurs seront linéairement dépendants. Ainsi tout $(d+2)$ -uplet de \mathcal{V} est support d'un circuit. Il y a donc exactement $2 \cdot \binom{n}{d+2}$ circuits si les points sont en position générale.

Dans le cas de points en deux dimensions, les supports des circuits sont de cardinal maximal 4. Un circuit de cardinal 2 correspond à des points confondus, un circuit de cardinal 3 correspond à des points alignés et un circuit de cardinal 4 correspond à des points en position générale (c'est-à-dire des points n'ayant pas de relation affine entre eux).

Dans le cas de points en trois dimensions, les circuits sont de cardinal maximal 5. Un circuit de cardinal 2 correspond à des points confondus, un circuit de cardinal 3 correspond à des points

alignés, un circuit de cardinal 4 correspond à des points coplanaires et un circuit de cardinal 5 correspond à des points en position générale.

Nous allons illustrer la notion de circuits sur l'exemple de la Figure 2.1. Comme les points sont en position générale, les circuits sont de cardinal 5. Il n'y a donc que deux circuits qui sont opposés. Ces deux circuits sont :

$$- + - + + \quad \text{et} \quad + - + - -$$

Le calcul de la moitié des circuits des configurations de points en deux dimensions de la Figure 1.2 nous donne le résultat suivant (l'autre moitié est obtenue en changeant tous les signes dans chacun des cocircuits) :

$$\begin{array}{lll} C_a(1\ 2\ 3) = + - + 0\ 0 & C_b(1\ 2\ 3) = + - + 0\ 0 & C_c(1\ 2\ 3) = + - + 0\ 0 \\ C_a(1\ 2\ 4\ 5) = - + 0 + - & C_b(1\ 2\ 4\ 5) = - + 0 + - & C_c(1\ 2\ 4\ 5) = - + 0 + - \\ C_a(1\ 3\ 4\ 5) = - 0 - - + & C_b(1\ 3\ 4\ 5) = + 0 - - + & C_c(1\ 3\ 4\ 5) = + 0 - - + \\ C_a(2\ 3\ 4\ 5) = 0 + + + - & C_b(2\ 3\ 4\ 5) = 0 - + + - & C_c(2\ 3\ 4\ 5) = 0 - + + - \end{array}$$

Les différences se situent cette fois-ci sur les deux derniers circuits. Le dernier circuit indique que le point 5 est à l'intérieur du triangle formé par les points 2, 3 et 4 dans la configuration (a) et à l'extérieur de ce triangle pour les autres configurations.

De même que pour les cocircuits, on peut calculer la moitié des circuits des configurations de points de la Figure 2.3. Le résultat est alors le suivant :

$$C_a(1\ 2\ 3\ 4\ 5) = - + - + + \quad C_b(1\ 2\ 3\ 4\ 5) = - + - + + \quad C_c(1\ 2\ 3\ 4\ 5) = - - + - +$$

Il est tout à fait normal de n'avoir qu'un seul circuit puisque nous sommes dans le cas de points en positions générales donc nous n'avons ni trois points alignés, ni quatre points coplanaires. Donc les supports des circuits sont les sous-ensembles à cinq éléments de \mathcal{E} .

Dualité. La théorie des matroïdes orientés possède la notion de dualité. Cette notion est valide dans les matroïdes orientés généraux mais ici nous explicitons la dualité dans le cas réalisable. Chaque matroïde orienté possède un matroïde orienté dual. Les circuits d'un matroïde orienté sont alors les cocircuits du matroïde orienté dual et inversement. On se servira de cette définition des circuits d'un matroïde orienté pour le calcul de ces circuits.

Nous allons voir comment calculer le matroïde orienté dual. Prenons le matroïde orienté associé à l'ensemble de points \mathcal{E} . Ce matroïde orienté est en fait calculé à partir de la matrice (v_1, \dots, v_n) qui est la matrice des vecteurs associés aux points de \mathcal{E} . Quitte à réordonner les vecteurs, on peut alors changer de base de référence pour écrire la matrice des vecteurs sous la forme $(I_{d+1} | A)$ où

I_{d+1} est la matrice identité de taille $(d+1) \times (d+1)$ et A est une matrice de taille $(d+1) \times (n-(d+1))$. Le matroïde orienté dual est donc le matroïde orienté associé aux vecteurs colonnes de la matrice $(-{}^t A | I_{n-(d+1)})$ où $-{}^t A$ est l'opposée de la matrice transposée de A et $I_{n-(d+1)}$ est la matrice identité de taille $(n-(d+1)) \times (n-(d+1))$.

Nous avons vu trois façons différentes (le chirotope, les cocircuits et les circuits) de définir le même matroïde orienté associé à une configuration de points. Dans la théorie des matroïdes orientés il existe des méthodes qui, à partir d'un de ces ensembles, permettent de déterminer les deux autres ensembles.

2.2 Propriétés géométriques codées par les matroïdes orientées

Maintenant que nous avons vu la définition, ainsi que l'application sur des exemples, de chacune des trois façons de caractériser un matroïde orienté, il est intéressant de voir quelles informations géométriques nous apporte chaque caractérisation. Pour plus d'informations sur le sujet, nous renvoyons à [20].

Orientation des simplexes

Nous avons vu que le chirotope d'une base nous indique l'orientation du simplexe correspondant aux points de la base. En reprenant l'exemple de la Figure 2.3, la base 2, 3, 4, 5 est positive pour les configurations (a) et (b). Cela signifie que dans ces deux configurations, les tétraèdres formés par les points 2, 3, 4, 5 ont la même orientation. Dans cet exemple la base de référence est directe. On a alors pour les configurations (a) et (b), qu'en se positionnant au point 2 et en regardant la face 3 4 5 dans le sens trigonométrique on voit les points dans l'ordre suivant : le point 3 puis le point 4 et enfin le point 5. Par contre la même base est négative dans la configuration (c). L'ordre des points en regardant la face depuis le point 5 est différent, c'est-à-dire que l'on verra le point 2 puis le point 4 et enfin le point 3. En changeant la base de référence, on change le signe de toutes les bases mais l'orientation du simplexe formé par les points ne change pas. Cela signifie que le signe d'une base ne nous permet pas de savoir quelle est l'orientation du tétraèdre mais uniquement de savoir si l'orientation de ce tétraèdre est la même que l'orientation de la base de référence. En comparant le signe de deux bases, on peut ainsi déterminer si les deux tétraèdres ont la même orientation ou non.

Enveloppe convexe

Nous avons vu dans la Section 2.1.4 que les cocircuits permettent de déterminer pour chaque hyperplan formé par des points de \mathcal{E} , la position des points de \mathcal{E} par rapport à cet hyperplan. Or les cocircuits permettent de déterminer les faces de l'enveloppe convexe de \mathcal{E} . Soit $\text{conv}(\mathcal{E})$ l'enveloppe convexe de \mathcal{E} . Un hyperplan de l'espace contient une face de $\text{conv}(\mathcal{E})$ (i.e. une face extrémale de dimension maximale) si et seulement si son complémentaire dans \mathcal{E} induit un cocircuit positif de M . Dans la Figure 2.3, on remarque que pour les configurations (a) et (b) on a $C^*(1\ 2\ 4) = 0\ 0\ -\ 0\ -$. Ainsi la face 1 2 4 est extrémale alors que ce n'est plus le cas dans la configuration (c).

Un élément e de \mathcal{E} est un point extrémal de $\text{conv}(\mathcal{E})$ si et seulement si il n'existe pas un circuit C de M avec $C^+ = \{e\}$. Les circuits peuvent donc nous permettre de savoir si un point est intérieur ou non à l'ensemble de points. En reprenant l'exemple de la Figure 2.1, tous les points appartiennent à l'enveloppe convexe. De manière équivalente, un élément e de \mathcal{E} est à l'intérieur de la structure formée par les points d'un sous-ensemble S de \mathcal{E} si et seulement si $S \cup \{e\}$ est support d'un circuit C avec $C^+ = \{e\}$.

2.3 Introduction d'une distance entre matroïdes orientés

Que ce soit aussi bien pour l'application à l'étude morphologique de structures anatomiques que pour la théorie purement mathématique, il est intéressant de pouvoir comparer les matroïdes orientés. En effet, pour ce qui est de l'application, il est important de pouvoir repérer les sujets proches les uns des autres ou à l'inverse de mettre en avant les sujets très différents des autres. L'étude de perturbations locales (léger déplacement d'un point par exemple), de mutations (inverser le signe d'une base dans un chirotope) et de suppressions et contractions d'éléments (notion de mineurs) ont déjà été étudiés pour créer un matroïde orienté à partir d'un matroïde orienté donné [5]. Ainsi ces modifications permettent de comparer des matroïdes orientés "proches" mais il semblerait que la comparaison de matroïdes orientés pouvant être arbitrairement éloignés n'ait pas encore été étudiée. Pour pouvoir comparer des matroïdes orientés associés à des configurations de points (qui peuvent être arbitrairement éloignés) il nous faut donc définir une distance entre deux matroïdes orientés.

Comme nous l'avons remarqué lors des préliminaires, un même matroïde orienté correspond à plusieurs configurations de points. On peut notamment prendre une configuration de points et regarder comment déplacer l'un de ces points sans changer le matroïde orienté associé. On remarque alors que ce point possède une zone de l'espace dans laquelle ce point peut être déplacé sans modifier le matroïde orienté. Cette zone est en fait délimitée par les hyperplans formés par les autres points. Cette remarque provient directement de la définition des cocircuits puisque si le point ne passe pas de l'autre côté d'un hyperplan alors les signes correspondant à ce point dans les cocircuits ne changent pas. Il en est de même pour les autres cocircuits (c'est-à-dire les cocircuits pour lesquels le point appartient à l'hyperplan). En effet, en déplaçant le point dans la zone on ne fait pas passer le point en question de l'autre côté des hyperplans, donc en déplaçant le point les hyperplans auxquels ce point appartient changent sans pour autant faire passer un point de l'autre côté d'un hyperplan.

Nous nous sommes alors intéressés à ce qui se produit comme changement dans le matroïde orienté lorsque le point quitte cette zone. Pour les mêmes raisons que précédemment, plusieurs cocircuits vont être modifiés. Il en est de même pour les circuits (dans le cas bien sûr où il y a assez de points pour avoir plusieurs circuits). Par contre, en position générale, lorsque le point passe de l'autre côté d'un hyperplan il n'y a qu'un seul chirotope de modifié. Ce chirotope correspond à la base formée par les points de l'hyperplan et le point qui a été déplacé. Si nous ne sommes pas en position générale, alors il se peut que plusieurs hyperplans soient confondus (dans le cas où il y a plus de d points qui appartiennent à l'hyperplan). Dans ce cas il y aura plusieurs chirotopes qui changeront en faisant passer un point d'un côté à l'autre de l'hyperplan.

distance
entre deux
matroïdes
orientés

Comme dans nos applications les points sont en position générale, il n'y aura qu'un chirotope qui change lorsqu'on fait passer un point d'un côté à l'autre d'un hyperplan, il est alors naturel de comparer deux matroïdes orientés par le nombre de chirotopes différents. On définit alors la *distance entre deux matroïdes orientés* comme étant égal au nombre de signes qui diffèrent entre les chirotopes de ces matroïdes orientés.

distance
mathéma-
tique

Pour que des matroïdes orientés soient comparables il faut que ces matroïdes orientés soient définis sur des configurations de points comparables. Pour cela, les configurations doivent comporter le même nombre de points pris dans un espace de même dimension. Les points devront également être ordonnés de la même façon (c'est-à-dire qu'il faudra faire une correspondance entre les points lors de leur numérotation) puisque l'on va comparer notamment $\chi(1 \ 2 \ \dots \ d+1)$ pour la première configuration avec $\chi(1 \ 2 \ \dots \ d+1)$ pour la deuxième configuration. De plus, le chirotope d'une base de référence devra être le même pour toutes les configurations pour que les signes des chirotopes indiquent bien la même orientation. On peut alors se demander si ce nombre de différences est bien une distance au sens mathématique du terme. On rappelle qu'une *distance mathématique* sur un ensemble E , est une application qui vérifie les propriétés suivantes :

$$\forall a, b \in E \quad d(a, b) = d(b, a) \quad (2.1)$$

$$\forall a, b \in E \quad d(a, b) = 0 \iff a = b \quad (2.2)$$

$$\forall a, b, c \in E \quad d(a, c) \leq d(a, b) + d(b, c) \quad (2.3)$$

Nous allons montrer que le nombre de différences entre deux chirotopes est bien une distance.

Lemme 2.1. *Soient M_1 et M_2 deux matroïdes orientés. Le nombre de différences entre le chirotope de M_1 et le chirotope de M_2 définit bien une distance mathématique.*

Preuve. Soient M_1, M_2 et M_3 trois matroïdes orientés. Notons $d(M_i, M_j)$ le nombre de chirotopes différents entre les matroïdes M_i et M_j . La propriété 2.1 étant triviale, nous allons montrer la propriété 2.2. Il est évident que $M_i = M_j \implies d(M_i, M_j) = 0$ puisque si les matroïdes orientés sont les mêmes cela signifie qu'ils ont les mêmes bases, les mêmes cocircuits et les mêmes circuits donc les chirotopes sont identiques. De même, si 2 matroïdes orientés ont les mêmes chirotopes, leurs ensembles des bases sont identiques et les matroïdes sont donc égaux ce qui prouve $d(M_i, M_j) = 0 \implies M_i = M_j$. La propriété 2.2 est donc vérifiée.

Il ne nous reste plus qu'à montrer l'inégalité triangulaire (la propriété 2.3). Nous allons tout d'abord le montrer pour un chirotope. Notons e un $(d+1)$ -uplet de vecteurs de V et $\chi_1(e)$, $\chi_2(e)$ et $\chi_3(e)$ les chirotopes des matroïdes orientés M_1 , M_2 et M_3 .

Posons $\Delta_e(i, j) = 1$ si $\chi_i(e) = \chi_j(e)$ et $\Delta_e(i, j) = 0$ sinon.

Ainsi si $\Delta_e(1, 3) = 0$ alors $\Delta_e(1, 2) \leq \Delta_e(1, 2) + \Delta_e(2, 3)$.

De même, si $\Delta_e(1, 3) = 1$ alors $\chi_1(e) \neq \chi_3(e)$. On a soit $\chi_1(e) \neq \chi_2(e)$ soit $\chi_2(e) \neq \chi_3(e)$ (voir les 2) donc on a soit $\Delta_e(1, 2) = 1$ soit $\Delta_e(2, 3) = 1$ (voir les 2). Du coup $\Delta_e(1, 3) \leq \Delta_e(1, 2) + \Delta_e(2, 3)$.

Or $d(M_i, M_j) = \sum_e \Delta_e(i, j)$. On a donc vérifier l'inégalité triangulaire $d(M_1, M_3) \leq d(M_1, M_2) + d(M_2, M_3)$ ce qui termine la preuve que le nombre de chirotopes différents entre 2 matroïdes orientés est bien une distance. \square

2.4 Avantages et limites pour le codage d'ensembles de points

Comme nous l'avons vu précédemment, le codage d'un ensemble de points par un matroïde orienté présente plusieurs avantages. Un premier avantage est que ce codage ne nécessite pas d'étape d'alignement ou de normalisation. On peut donc directement coder un ensemble de points par un matroïde orienté sans avoir à utiliser des méthodes pouvant créer des erreurs de calculs.

Un autre avantage de ce codage est qu'il est combinatoire (en d'autres termes, il utilise un nombre fini de signes contrairement aux codages numériques) ce qui fait qu'on peut stocker simplement et de manière exacte ce codage sur un ordinateur. De plus la distance entre matroïde que nous venons de définir se prête bien à une utilisation par ordinateur puisque les valeurs des distances sont des entiers. Dans le calcul et l'utilisation des distances n'y a donc pas de problèmes d'imprécision que l'on rencontre lors de calculs avec des flottants [59].

Par contre ce codage présente également des inconvénients. Un premier inconvénient de ce codage est qu'il peut être sensible à de faibles variations sur la position de points. En effet, si un point est proche d'un ou plusieurs hyperplans formés par les autres points alors un déplacement même très faible peut positionner le point sur l'hyperplan ou le faire basculer de l'autre côté de l'hyperplan. Un tel déplacement aurait pour conséquence respectivement de transformer le signe en 0 ou de l'inverser.

Un autre inconvénient concerne la taille d'un chirotope (c'est-à-dire le nombre de signes). En effet, pour coder un ensemble de n points en dimension d on associe un signe à chaque $(d+1)$ -uplet de points. Le chirotope associé à un tel ensemble de points est alors un vecteur de taille $\binom{n}{d+1}$. Dans nos applications d sera égal à 3 mais même avec cette valeur pour d , la taille d'un vecteur peut poser problème si n est grand.

Lors de nos premières expérimentations, nous nous sommes aperçu que la majorité des $(d+1)$ -uplets de points ont le même signe pour tous les échantillons. Nous avons choisi d'appeler une telle base une *base fixe*. Nous étudierons ces bases fixes dans la Partie II. Dans la Partie III nous utiliserons les bases non-fixes afin de classifier les échantillons et de caractériser ces classes.

base fixe



Partie II. Bases fixes issues d'ordres sur les coordonnées des points

Dans la seconde partie de cette thèse, nous nous intéressons aux bases fixes. Nous avons vu dans la Section 2.3 qu'une grande partie des bases sont fixes. En fait, nous travaillons avec des points de repère anatomiques ce qui implique des relations entre ces points que nous pouvons formaliser par des contraintes d'ordre sur leurs coordonnées. Par exemple, si l'on considère les deux oreilles internes (notées OG pour l'oreille interne gauche et OD pour l'oreille interne droite), la pointe du menton (noté M) et le milieu du front (noté F) d'un crâne alors OD sera le point le plus à droite et OG sera le point le plus à gauche, et ce, quel que soit le crâne et la variabilité naturelle. En notant x l'axe allant de la droite à la gauche du crâne, ceci se formalise par :

$$x_{OD} < x_M < x_{OG} \text{ et } x_{OD} < x_F < x_{OG}$$

La question qui se pose alors est : est-ce que ces contraintes d'ordre impliquent que la base (OG OD M F) sera toujours fixe ? Plus généralement, si on prend une des $\binom{n}{4}$ bases et si on fixe un ordre entre ses quatre points pour chaque coordonnée, est-ce que cela implique au la base est fixe. On obtient alors trois relations d'ordre et l'on cherche à déterminer s'ils impliquent que la base est fixe ou si cette base peut avoir des signes différents.

Ce problème peut également être posé sous forme de matrices puisque nous avons vu dans la Section 2.1 que le signe d'une base est le signe d'un déterminant d'une matrice formée par les coordonnées des points. Dans ce cas, les ordres sont sur les coefficients de la matrice et chaque ordre est associé à une ligne de la matrice. Le problème est alors de déterminer si pour tous les coefficients respectant ces ordres, le signe du déterminant de la matrice est constant (ce qui est équivalent à ce que le déterminant de la matrice soit non-nul).

Dans cette partie nous nous intéressons à des bases en dimension quelconque pour lesquels, pour chaque coordonnée de l'espace, un ordre est fixé entre les différents sommets. Dans le Chapitre 3, nous cherchons à déterminer l'orientation de tels simplexes uniquement à partir des ordres donnés. Nous expliquons qu'il existe des transformations d'ensembles d'ordres tels qu'un ensemble d'ordres e et un ensemble d'ordres e' obtenu par transformation de e , sont équivalents pour ce problème. En d'autres termes, un ensemble d'ordres e implique qu'une base B est fixe

si et seulement si un ensemble d'ordres e' obtenu par transformation de e implique que B est fixe. Nous proposons un algorithme combinatoire pour les cas en dimension deux et trois, qui permet de savoir si, étant donnée un ensemble d'ordres, le déterminant sera de signe constant quelles que soient les coordonnées des points.

Dans le Chapitre 4, nous nous intéressons au nombre d'ensembles d'ordres totaux à transformations près en fonction de la dimension. Nous expliquons comment fonctionne le programme qui nous a permis de calculer le nombre d'ensemble d'ordres totaux à transformations près lorsque la dimension est inférieure ou égale à six.

Characterization in dimension 2 and 3 and conjecture in higher dimensions

Ce chapitre reprend l'Article [21] qui a été soumis à la revue *Linear Algebra and its Applications* en mai 2013.

In this chapter, we address the problem of testing when orderings on coordinates of n points in an $(n - 1)$ -dimensional affine space, one ordering for each coordinate, suffice to determine if these points are the vertices of a simplex (i.e. are affinely independent), and the orientation of this simplex, independently of the real values of the coordinates. In other words, we want to know when the sign (or the non-nullity) of the determinant of a matrix whose columns correspond to affine points is determined by orderings given on the values on each row. We completely solve the problem in dimensions 2 and 3, providing a direct combinatorial characterization, together with a formal calculus method, that can be seen also as a decision algorithm, and which relies on testing the existence of a suitable inductive cofactor expansion of the determinant. We conjecture that the method we use generalizes in higher dimensions. A motivation for this work is to be part of a study on how oriented matroids encode shapes of 3-dimensional objects, with applications in particular to the analysis of anatomical data for physical anthropology and clinical research.

Introduction

We consider n points in an $(n - 1)$ -dimensional real affine space. For each of the $n - 1$ coordinates, an ordering is given, applied on the n values of the points with respect to this coordinate. We address the problem of testing if these points are the vertices of a simplex (i.e. are affinely independent, i.e. do not belong to a same hyperplane), and of determining the orientation of this simplex, assuming only that their coordinates satisfy the given orderings, independently of their real values.

More formally, we consider the following generic matrix (where each e_i is the label of a point

and each b_i is the index of a coordinate)

$$M_{\mathcal{E}, \mathcal{B}} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_{e_1, b_1} & x_{e_2, b_1} & \dots & x_{e_n, b_1} \\ x_{e_1, b_2} & x_{e_2, b_2} & \dots & x_{e_n, b_2} \\ \vdots & \vdots & & \vdots \\ x_{e_1, b_{n-1}} & x_{e_2, b_{n-1}} & \dots & x_{e_n, b_{n-1}} \end{pmatrix}$$

together with orderings given on the values on each row, and we want to know when the sign (or the non-nullity) of its determinant is determined by these orderings only.

Equivalently, we consider the above formal matrix and the affine algebraic variety of $\mathbb{R}^{n \times (n-1)}$ whose equation is $\det(M_{\mathcal{E}, \mathcal{B}}) = 0$. Then we look for which (open) regions of $\mathbb{R}^{n \times (n-1)}$ delimited by the hyperplanes $x_{e_i, b_k} = x_{e_j, b_k}$ for all $1 \leq i, j \leq n$ and all $1 \leq k \leq n-1$, have a non-empty intersection with this variety (obviously, regions delimited by these hyperplanes are in canonical bijection with coordinate linear orderings).

In this paper, we completely solve the problem in dimensions 2 (Section 3.3.1) and 3 (Section 3.3.2), providing a direct combinatorial characterization, together with a combinatorial formal calculus method, that can be seen also as a decision algorithm, to test if the orientation is determined or not. More precisely, our method relies on testing the existence of a suitable inductive cofactor expansion of the determinant, from which a combinatorial formal calculus is able to determine the sign of the determinant. We conjecture that this formal characterization generalizes in higher dimensions (Section 3.2).

We mention that the problem addressed here is formally close to the classical problem of sign nonsingular matrices (SNS), see [10], but the two problems are quite independent. Let us explain this roughly. The common feature of the two problems relies in the following case. Consider a square $n \times n$ matrix N whose entries are signs in $\{+, -\}$. To us, such a $n \times n$ matrix N can be obtained naturally from $M_{\mathcal{E}, \mathcal{B}}$ and a linear ordering at each row by subtracting a column - say e_i - to every other column, and replacing entries with their signs w.r.t. the linear orderings; thus, conversely, in our setting, the sign data in such a matrix N is interpreted as an ordering relation at each row of type $:(\text{the set of } -) < (\text{the set of } +)$, corresponding to $:(\text{a set } A \subset \mathcal{E}) < e_i < (\text{a set } B \subset \mathcal{E})$. The question, in both settings, is: is this matrix N invertible for every possible real values chosen as entries of the matrix, provided these values have same signs as the signs in the matrix? To this particular question, the answer is always NO, whatever are the signs (unless $n \leq 2$). But, in more general settings, the answer can be YES. The SNS setting and ours consist in two different variants of this question, in which this question is non-trivial, and provides interesting classes of sign patterns. In the SNS setting, the refinement is to consider the same question with signs in $\{+, -, 0\}$ instead of $\{+, -\}$. In our setting, the refinement is to consider the same question, with signs in $\{+, -\}$, but with the available real values restricted to values satisfying more involved ordering relations at each row (between all elements and not only between two subsets A and B). There seem to be no obvious connection between those two problems since the zeros in the SNS setting and the linear orderings in our setting yield constraints of quite different natures to the sets of real values to be tested.

Finally, we point out that a motivation for this work is to be part of a general study on how oriented matroids [5] encode shapes of 3-dimensional objects, with applications in particular to the analysis of anatomical data for physical anthropology and clinical research [24][23]. In these

applications, we usually study a set of models belonging to a given group (e.g. a set of 3D landmark points located on human or primate skulls) and we look for the significant properties encoded by the combinatorial structure. The above result allows us to distinguish chirotopes (i.e. simplex orientations) which are determined by the “generic” form (e.g. in any skull, the mouth is below the eyes) from those which are specific to anatomical variations. As an example, some results on anatomical 3D data are presented in Sections 3.1.2 and 3.3.3.

3.1 Preliminaries

3.1.1 Formalism and terminology of the problem

We warn the reader that we use on purpose a rather abstract formalism throughout the paper (formal variables instead of real values, indices within arbitrary ordered sets instead of integers). This will allow us to get easier and non-ambiguous constructions and definitions.

Let us fix an (ordered) set $\mathcal{E} = \{e_1, \dots, e_n\}$, with size n , of *labels*, and an (ordered) canonical basis $\mathcal{B} = \{b_1, \dots, b_{n-1}\}$, with size $n-1$, of the $(n-1)$ -dimensional real space \mathbb{R}^{n-1} . We denote $M_{\mathcal{E}, \mathcal{B}}$ - or M for short when the context is clear - the formal matrix

$$M_{\mathcal{E}, \mathcal{B}} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_{e_1, b_1} & x_{e_2, b_1} & \dots & x_{e_n, b_1} \\ x_{e_1, b_2} & x_{e_2, b_2} & \dots & x_{e_n, b_2} \\ \vdots & \vdots & & \vdots \\ x_{e_1, b_{n-1}} & x_{e_2, b_{n-1}} & \dots & x_{e_n, b_{n-1}} \end{pmatrix}$$

whose entry at column i and row $j+1$, for $1 \leq i \leq n$ and $1 \leq j \leq n-1$, is the formal variable x_{e_i, b_j} . The determinant $\det(M_{\mathcal{E}, \mathcal{B}})$ of this formal matrix is a multivariate polynomial on these formal variables, and the main object studied in this paper.

Let \mathcal{P} be a set of n points, labeled by \mathcal{E} , in \mathbb{R}^{n-1} considered as an affine space. We denote $M_{\mathcal{E}, \mathcal{B}}(\mathcal{P})$ - or $M(\mathcal{P})$ for short - the matrix whose columns give the coordinates of points in \mathcal{P} w.r.t. the basis \mathcal{B} , that is specifying real values for the formal variables x_{e_i, b_j} in the matrix $M_{\mathcal{E}, \mathcal{B}}$ above. For $e \in \mathcal{E}$ and $b \in \mathcal{B}$, we denote $x_{e, b}(\mathcal{P})$ the real value given to the formal variable $x_{e, b}$ in \mathcal{P} . We may sometimes denote $x_{e, b}$ for short instead of $x_{e, b}(\mathcal{P})$ when the context is clear. We call *orientation* of \mathcal{P} , or *chirotope* of \mathcal{P} in the oriented matroid terminology, the sign of $\det(M(\mathcal{P}))$, belonging to the set $\{+, -, 0\}$. It is the sign of the real evaluation of the polynomial $\det(M)$ at the real values given by \mathcal{P} . This sign is not equal to zero if and only if \mathcal{P} forms a *simplex* (basis of the affine space).

We call *ordering configuration* on $(\mathcal{E}, \mathcal{B})$ - or *configuration* for short - a list \mathcal{C} of $n-1$ orderings $<_{b_1}, \dots, <_{b_{n-1}}$ on \mathcal{E} , one ordering for each element of \mathcal{B} . In general, such an ordering can be any partial ordering. If every ordering $<_b$, $b \in \mathcal{B}$, is linear, then \mathcal{C} is called a *linear ordering configuration*. An element of \mathcal{E} which is the smallest or the greatest in a linear ordering on \mathcal{E} is called *extreme* in this ordering. We call *reversion* of an ordering the ordering obtained by reversing every inequality in this ordering.

Given a configuration \mathcal{C} on $(\mathcal{E}, \mathcal{B})$ and a set of n points \mathcal{P} labeled by \mathcal{E} , we say that \mathcal{P} *satisfies* \mathcal{C} if, for all $b \in \mathcal{B}$, the natural order (in the set of real numbers \mathbb{R}) of the coordinates b of the points

orientation
chirotope

simplex
configuration

linear ordering
configuration
extreme
reversion
satisfies

in \mathcal{P} is compatible with the ordering $<_b$ of \mathcal{C} , that is precisely :

$$\forall b \in \mathcal{B}, \forall e, f \in \mathcal{E}, e <_b f \Rightarrow x_{e,b}(\mathcal{P}) < x_{f,b}(\mathcal{P}).$$

One may observe that the set of all \mathcal{P} satisfying \mathcal{C} forms a convex polyhedron, more precisely : a (full dimensional) region of the space $\mathbb{R}^{n \times (n-1)}$, delimited by some hyperplanes of equations of type $x_{e,b} = x_{f,b}$ for $b \in \mathcal{B}$ and $e, f \in \mathcal{E}$.

fixed
sign of
 $\det(M)$
non-fixed

We say that a configuration \mathcal{C} is *fixed* if all the sets of points \mathcal{P} satisfying \mathcal{C} form a simplex and have the same orientation. In this case, the sign of $\det(M(\mathcal{P}))$ is the same for all \mathcal{P} satisfying \mathcal{C} . Then we call *sign of $\det(M)$* this sign, belonging to $\{+, -\}$ accordingly, and we denote it $\sigma_{\mathcal{C}}(\det(M))$. If \mathcal{C} is *non-fixed*, then its *sign* is $\sigma_{\mathcal{C}}(\det(M)) = \pm$.

Lemma 3.1. *The following propositions are equivalent :*

- (a) *The configuration \mathcal{C} is non-fixed, that is $\sigma_{\mathcal{C}}(\det(M)) = \pm$.*
- (b) *There exist two sets of points \mathcal{P}_1 and \mathcal{P}_2 satisfying \mathcal{C} and forming simplices that do not have the same orientation, that is $\det(M(\mathcal{P}_1)) > 0$ and $\det(M(\mathcal{P}_2)) < 0$;*
- (c) *There exists a set of points \mathcal{P} satisfying \mathcal{C} and such that the points of \mathcal{P} belong to one hyperplane, that is $\det(M(\mathcal{P})) = 0$.*

Preuve. By definition we have a) if and only if b) or c). The region of the space $\mathbb{R}^{n \times (n-1)}$ whose elements \mathcal{P} satisfy \mathcal{C} is a convex and, topologically, an open set of points in $\mathbb{R}^{n \times (n-1)}$. So we have that b) implies c) by convexity and by continuity of the determinant And we have that c) implies b) since, given \mathcal{P} in this region such that $\det(M(\mathcal{P})) = 0$, one can add a matrix small enough to $M(\mathcal{P})$ to get \mathcal{P}' in the same region and such that $\det(M(\mathcal{P}')) > 0$, or also such that $\det(M(\mathcal{P}')) < 0$. \square

equivalent
configurations

Two configurations on $(\mathcal{E}, \mathcal{B})$ are called *equivalent* if they are equal up to a permutation of \mathcal{B} , a permutation of \mathcal{E} (relabelling), and some reversions of orderings (symmetries from the geometrical viewpoint). Note that changing a configuration into an equivalent one comes, in a matricial setting, to change the orderings of rows, of columns, and to multiply some rows by -1 . Obviously those operations do not change the non-nullity of the determinant, hence two equivalent configurations are fixed or non-fixed simultaneously.

Now, given an ordering configuration \mathcal{C} , the aim of the paper is to determine if \mathcal{C} is fixed or non-fixed.

3.1.2 An example from applications

Let us consider ten anatomical landmark points in \mathbb{R}^3 chosen by experts on the 3D model of a skull from [8], as shown on Figure 3.1. We choose a canonical basis $(O, \vec{x}, \vec{y}, \vec{z})$ such that the axis \vec{x} goes from the right of the skull to its left, the axis \vec{y} goes from the bottom of the skull to its top, and the axis \vec{z} goes from the front of the skull to its back. The specificity of this 3D model as being a skull implies that some coordinate ordering relations are satisfied by those points : for instance the point 9 (right internal ear) will always be on the right, above and behind with respect to point 5 (right part of the chin). Figures 3.2 and 3.3 show respectively those points from the front and from the right of the model, together with a grid representing those coordinate ordering relations.

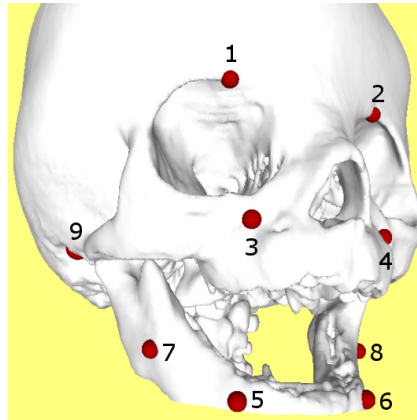


FIGURE 3.1 – Ten anatomic points on a skull model [8]

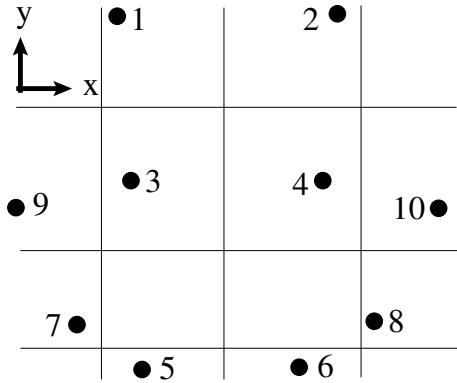


FIGURE 3.2 – View from the front

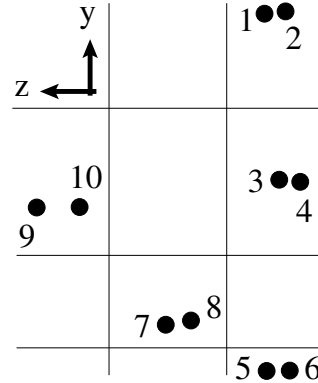


FIGURE 3.3 – View from the right

We are usually (e.g. in [24]) given a set of such models, coming from various individuals (with possibly some pathologies) and species (e.g. primates and humans), by experts of those fields who are interested in characterizing and classifying mathematically those models. In this paper, our aim is to detect which configurations are fixed, independently of the real values of the landmarks, meaning that the relative positions of points satisfying these configurations do not depend on some anatomical variabilities (e.g. being a primate or a human skull), but just on the generic shape of the model (i.e. being a skull).

The ordering configurations are represented on Figures 3.2 and 3.3, with \mathcal{E} any set of four points, and \mathcal{B} corresponding to the three axis $\{x, y, z\}$. As a preliminary example, let us consider the relations between points labelled by $\mathcal{E} = \{1, 5, 9, 10\}$, we get the following configuration \mathcal{C} , as illustrated on Figures 3.4, 3.5 and 3.6 :

$$\begin{aligned}
 &9 <_x 1 <_x 10 \quad \text{and} \quad 9 <_x 5 <_x 10 \\
 &5 <_y 9 <_y 1 \quad \text{and} \quad 5 <_y 10 <_y 1 \\
 &1 <_z 9 \quad \text{and} \quad 1 <_z 10 \quad \text{and} \quad 5 <_z 9 \quad \text{and} \quad 5 <_z 10
 \end{aligned}$$

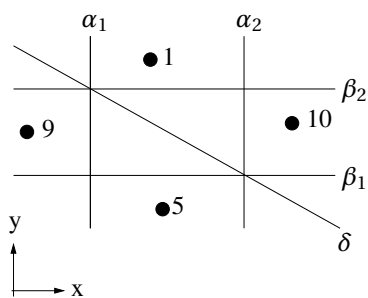


FIGURE 3.4 – View from the front

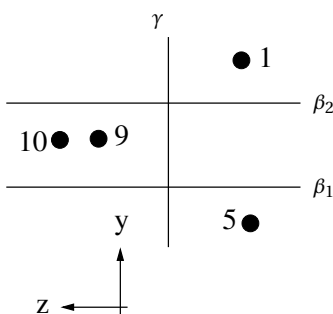


FIGURE 3.5 – View from the left

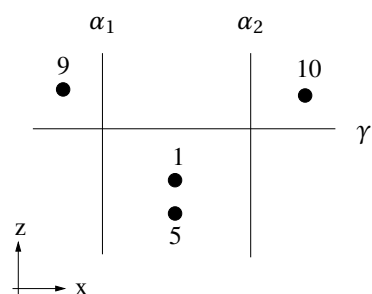


FIGURE 3.6 – View from below

We leave as an easy geometrical exercise to prove that this configuration is fixed. The idea is the following. Let us prove that the line (1,5) and the line (9,10) cannot intersect, which proves that the points cannot belong to a same hyperplane, and hence form a simplex with fixed orientation. Consider two planes α_1, α_2 parallel to the directions y, z , two planes β_1, β_2 parallel to the directions x, z , and one plane γ parallel to the directions x, y , consistent with the coordinate orderings, as shown on Figures 3.4, 3.5 and 3.6. These 5 planes separate \mathbb{R}^3 in 18 regions. Consider the plane δ containing the intersections of planes $\alpha_1, \beta_2, \gamma$ and $\alpha_2, \beta_1, \gamma$, and parallel to the direction z , as shown on Figure 3.4. Consider a region (among the 18 regions) intersecting δ . Prove that, if the line (1,5) and the line (9,10) both intersect this region, then the two intersections are contained in two distinct parts of this region separated by δ , meaning that the two lines do not intersect. The other cases (other regions) are either symmetric to this one or trivial.

In the rest of the paper, we develop tools to detect automatically fixed configurations, without using specific geometrical constructions for each configuration as done above, but rather by unifying all configurations under a common combinatorial criterion. We will continue to study this example under this approach in Section 3.3.3.

3.2 Computable fixity criteria and conjectures

3.2.1 From partial orderings to linear orderings

We recall that a linear extension of an ordering on a set \mathcal{E} is a linear ordering on \mathcal{E} compatible with this ordering. A *linear extension* of an ordering configuration \mathcal{C} on $(\mathcal{E}, \mathcal{B})$ is a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$ obtained by replacing each ordering on \mathcal{E} in \mathcal{C} by one of its linear extensions.

Lemma 3.2. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$. If there exists a set \mathcal{P} of n points satisfying \mathcal{C} and contained in a hyperplane, then there exists a set of n points \mathcal{P}' contained in a hyperplane and a linear extension \mathcal{C}' of \mathcal{C} satisfied by \mathcal{P}' .*

Preuve. Assume that, for every row of $M(\mathcal{P})$ except the first, all entries in this row are distinct. Then the linear orderings on these real values at each row define a linear ordering configuration \mathcal{C}' . This

linear
extension

configuration \mathcal{C}' is a linear extension of \mathcal{C} since \mathcal{P} satisfies \mathcal{C} . Then $\mathcal{P}' = \mathcal{P}$ and \mathcal{C}' have the required properties.

Assume two columns with labels $e, f \in \mathcal{E}$ of $M(\mathcal{P})$ are equal. Let $v = (v_1, \dots, v_{n-1})$ be a vector parallel to the (affine) hyperplane containing \mathcal{P} . Let \mathcal{P}' be obtained by adding $\varepsilon \cdot (0, v_1, \dots, v_{n-1})$ to coordinates of the point labelled by f , for some $\varepsilon > 0$. Obviously, the value ε can be chosen small enough to have that \mathcal{P}' still satisfies \mathcal{C} . By definition of v , \mathcal{P}' is contained in the same hyperplane as \mathcal{P} . Using this construction iteratively finally yields a set of points \mathcal{P}' satisfying \mathcal{C} , contained in the same hyperplane as \mathcal{P} and such that $M(\mathcal{P}')$ has all its columns distinct.

Assume that in the row $b \in \mathcal{B}$ in $M(\mathcal{P})$, the values at columns labelled by $e, f \in \mathcal{E}$ are the same. Up to transforming \mathcal{P} as above, we assume that those two columns are not equal. Then there exists a row $b' \in \mathcal{B}$ such that $x_{e,b'} \neq x_{f,b'}$. Let \mathcal{P}' be the set of points whose matrix $M(\mathcal{P}')$ is obtained by adding ε times row b' to row b in $M(\mathcal{P})$, for some $\varepsilon > 0$. Obviously, the value ε can be chosen small enough to have that \mathcal{P}' still satisfies \mathcal{C} . Since the determinant of the matrices $M(\mathcal{P})$ and $M(\mathcal{P}')$ are equal, \mathcal{P}' is contained in a hyperplane. Using this construction iteratively finally yields a set of points \mathcal{P}' satisfying \mathcal{C} , contained in a hyperplane and satisfying the hypothesis of the first paragraph of this proof. \square

Proposition 3.3. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$. The configuration \mathcal{C} is non-fixed if and only if there exists a non-fixed linear extension of \mathcal{C} . The configuration \mathcal{C} is fixed if and only if every linear extension of \mathcal{C} is fixed.*

Preuve. We prove the first assertion in the proposition. The second is obviously equivalent. By Lemma 3.1, if \mathcal{C} is non-fixed then there exists a set of n points \mathcal{P} satisfying \mathcal{C} such that $\det(M(\mathcal{P})) = 0$, that is \mathcal{P} is contained in a hyperplane. Lemma 3.2 implies that there exists \mathcal{P}' satisfying a linear extension \mathcal{C}' of \mathcal{C} and such that $\det(M(\mathcal{P}')) = 0$. Hence \mathcal{C}' is non-fixed. Conversely, let \mathcal{C}' be a non-fixed linear extension of \mathcal{C} . By Lemma 3.1, there exists \mathcal{P} such that $\det(M(\mathcal{P})) = 0$ and \mathcal{P} satisfies \mathcal{C}' . In particular, \mathcal{P} satisfies \mathcal{C} , and hence \mathcal{C} is non-fixed. \square

The above result allows to test only the fixity of linear ordering configurations to deduce the fixity of any configuration. In what follows, we will concentrate on linear ordering configurations.

3.2.2 Formal fixity

Let \mathcal{C} be a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$. We consider formal expression of type $x_{e,b} - x_{f,b}$ for $e, f \in \mathcal{E}$, $e \neq f$, and $b \in \mathcal{B}$, which we may sometimes denote $x_{e-f,b}$ for short. Such a formal expression gets a *formal sign w.r.t. \mathcal{C}* denoted $\sigma_{\mathcal{C}}(x_{e,b} - x_{f,b})$ and belonging to $\{\boxed{+}, \boxed{-}\}$, the following way :

$$\sigma_{\mathcal{C}}(x_{e,b} - x_{f,b}) = \boxed{+} \text{ if } f <_b e; \quad \sigma_{\mathcal{C}}(x_{e,b} - x_{f,b}) = \boxed{-} \text{ if } e <_b f.$$

Recall that the polynomial $\det(M_{\mathcal{E}, \mathcal{B}})$ is a multivariate polynomial on variables $x_{e,b}$ for $b \in \mathcal{B}$ and $e \in \mathcal{E}$. Assume a particular formal expression of $\det(M_{\mathcal{E}, \mathcal{B}})$ is a sum of multivariate monomials where each variable is replaced by some $x_{e,b} - x_{f,b}$, for $b \in \mathcal{B}$ and $e, f \in \mathcal{E}$. Various expressions of this type can be obtained by suitable transformations and determinant cofactor expansions from the matrix M , as we will do more precisely below. This particular expression of $\det(M_{\mathcal{E}, \mathcal{B}})$ gets a

formal sign

formal sign w.r.t. \mathcal{C} belonging to $\{\boxed{+}, \boxed{-}, \boxed{?}\}$, by replacing each expression of type $x_{e,b} - x_{f,b}$ with its formal sign $\sigma_{\mathcal{C}}(x_{e,b} - x_{f,b})$ and applying the following formal calculus rules :

$$\begin{aligned} \boxed{+} \cdot \boxed{+} &= \boxed{-} \cdot \boxed{-} = \boxed{+}, & \boxed{+} \cdot \boxed{-} &= \boxed{-} \cdot \boxed{+} = \boxed{-}, \\ \boxed{+} + \boxed{+} &= \boxed{+} - \boxed{-} = \boxed{+}, & \boxed{-} + \boxed{-} &= \boxed{-} - \boxed{+} = \boxed{-}, \\ \boxed{+} + \boxed{-} &= \boxed{-} + \boxed{+} = \boxed{?}, \end{aligned}$$

and the result of any operation involving a $\boxed{?}$ term or factor is also $\boxed{?}$.

We say that \mathcal{C} is *formally fixed* if $\det(M_{\mathcal{E},\mathcal{B}})$ has such a formal expression whose formal sign is not $\boxed{?}$.

formally
fixed

Example. Consider the following matrix $M = M_{\mathcal{E},\mathcal{B}}$ for $\mathcal{E} = \{a, b, c\}$ and $\mathcal{B} = \{1, 2\}$:

$$M = \begin{pmatrix} 1 & 1 & 1 \\ x_{a,1} & x_{b,1} & x_{c,1} \\ x_{a,2} & x_{b,2} & x_{c,2} \end{pmatrix}$$

and consider the configuration \mathcal{C} defined by :

$$\begin{array}{ccc} a <_1 & b <_1 & c \\ b <_2 & c <_2 & a \end{array}$$

A formal expression of $\det(M)$ is :

$$\det(M) = x_{b-a,1} \cdot x_{c-a,2} - x_{b-a,2} \cdot x_{c-a,1}$$

whose formal sign w.r.t. \mathcal{C} is

$$\boxed{+} \cdot \boxed{-} - \boxed{-} \cdot \boxed{+} = \boxed{?}.$$

Another formal expression of $\det(M)$ is :

$$\det(M) = x_{b-a,1} \cdot x_{c-b,2} - x_{b-a,2} \cdot x_{c-b,1}$$

whose formal sign w.r.t. \mathcal{C} is

$$\boxed{+} \cdot \boxed{+} - \boxed{-} \cdot \boxed{+} = \boxed{+}.$$

This second expression shows that \mathcal{C} is formally fixed.

Observation 3.4. *If \mathcal{C} is formally fixed, then \mathcal{C} is fixed.*

More precisely, given an expression as above whose formal sign w.r.t. \mathcal{C} is $\boxed{+}$ or $\boxed{-}$, the evaluation of this determinant for any set of real values \mathcal{P} satisfying \mathcal{C} necessarily provides a real number whose sign is consistent with the formal sign of this expression. In this case, this resulting sign does not depend on the chosen expression as soon as it is not $\boxed{?}$, and $\sigma_{\mathcal{C}}(\det(M))$ equals this sign.

Conversely, one may wonder if for every fixed configuration there would exist a suitable expression of the determinant showing formally that \mathcal{C} is fixed by this way. That is, equivalently : if every formal expression of $\det(M_{\mathcal{E},\mathcal{B}})$ has formal sign $\boxed{?}$, then $\sigma_{\mathcal{C}}(\det(M)) = \boxed{\pm}$. We strongly believe in this result, which we state as a conjecture, and which we will prove for $n \leq 4$ (see Theorems 3.8 and 3.10).

Conjecture 1. *Let \mathcal{C} be a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$. Then \mathcal{C} is fixed if and only if \mathcal{C} is formally fixed.*

3.2.3 Formal fixity by expansion

Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$, and $\mathcal{E}' = \mathcal{E} \setminus \{e\}$, $\mathcal{B}' = \mathcal{B} \setminus \{b\}$ for some $e \in \mathcal{E}$, $b \in \mathcal{B}$. We call *configuration induced by \mathcal{C} on $(\mathcal{E}', \mathcal{B}')$* the configuration on $(\mathcal{E}', \mathcal{B}')$ obtained by restricting every ordering $<_{b'}$, $b' \in \mathcal{B}'$, of \mathcal{C} to \mathcal{E}' . Moreover, we say that *all the configurations induced by \mathcal{C} on \mathcal{E}' are fixed* if, for every $b \in \mathcal{B}$, the configuration induced by \mathcal{C} on $(\mathcal{E}', \mathcal{B} \setminus \{b\})$ is a fixed configuration. Note that, from the geometrical viewpoint, if \mathcal{P} is a set of points satisfying \mathcal{C} , and \mathcal{P}_e is obtained by removing from \mathcal{P} the point with label $e \in \mathcal{E}$, then the projection \mathcal{P}' of \mathcal{P}_e on \mathcal{B}' along b satisfies \mathcal{C}' . Indeed, the matrix $M_{\mathcal{E}', \mathcal{B}'}$, resp. $M_{\mathcal{E}', \mathcal{B}'}(\mathcal{P}')$, is obtained by removing the column corresponding to e and the row corresponding to b from $M_{\mathcal{E}, \mathcal{B}}$, resp. $M_{\mathcal{E}, \mathcal{B}}(\mathcal{P})$.

configuration induced by \mathcal{C} all the configurations induced by \mathcal{C} are fixed

Let $M = M_{\mathcal{E}, \mathcal{B}}$ as previously, with $\mathcal{E} = \{e_1, \dots, e_n\}_<$ and $\mathcal{B} = \{b_1, \dots, b_{n-1}\}_<$. Let $e_i, e_j \in \mathcal{E}$, with $e_i \neq e_j$. Consider the matrix obtained from M by subtracting the j -th column (corresponding to e_j), from the i -th column (corresponding to e_i), that is :

$$\begin{pmatrix} 1 & \dots & 1 & 0 & 1 & \dots & 1 \\ x_{e_1, b_1} & \dots & x_{e_{i-1}, b_1} & x_{e_i, b_1} - x_{e_j, b_1} & x_{e_{i+1}, b_1} & \dots & x_{e_n, b_1} \\ x_{e_1, b_2} & \dots & x_{e_{i-1}, b_2} & x_{e_i, b_2} - x_{e_j, b_2} & x_{e_{i+1}, b_2} & \dots & x_{e_n, b_2} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ x_{e_1, b_{n-1}} & \dots & x_{e_{i-1}, b_{n-1}} & x_{e_i, b_{n-1}} - x_{e_j, b_{n-1}} & x_{e_{i+1}, b_{n-1}} & \dots & x_{e_n, b_{n-1}} \end{pmatrix}$$

The determinant of this matrix equals $\det(M)$. The cofactor expansion formula for the determinant of this matrix with respect to its i -th column yields :

expression of $\det(M)$ by expansion

$$\det(M_{\mathcal{E}, \mathcal{B}}) = \sum_{k=1}^{n-1} (-1)^{i+k+1} \cdot (x_{e_i, b_k} - x_{e_j, b_k}) \cdot \det(M_{\mathcal{E} \setminus \{e_i\}, \mathcal{B} \setminus \{b_k\}})$$

which we call *expression of $\det(M)$ by expansion with respect to (e_i, e_j)* .

Then the above particular expression of $\det(M)$ gets a *formal sign* w.r.t. \mathcal{C} the following way. First, replace each expression of type $x_{e, b} - x_{f, b}$ with its formal sign w.r.t. \mathcal{C} in $\{\boxed{+}, \boxed{-}\}$, and replace each $\det(M_{\mathcal{E} \setminus \{e_i\}, \mathcal{B} \setminus \{b_k\}})$, $1 \leq k \leq n-1$, with its sign $\sigma_{\mathcal{C}_k}(\det(M_{\mathcal{E} \setminus \{e_i\}, \mathcal{B} \setminus \{b_k\}})) \in \{\boxed{+}, \boxed{-}, \boxed{\pm}\}$, where \mathcal{C}_k is the configuration induced by \mathcal{C} on $(\mathcal{E} \setminus \{e_i\}, \mathcal{B} \setminus \{b_k\})$. This leads to the formal expression :

$$\sum_{k=1}^{n-1} (-1)^{i+k+1} \cdot \sigma_{\mathcal{C}}(x_{e_i, b_k} - x_{e_j, b_k}) \cdot \sigma_{\mathcal{C}_k}(\det(M_{\mathcal{E} \setminus \{e_i\}, \mathcal{B} \setminus \{b_k\}})),$$

Then, provide the formal sign of this expression by using the same formal calculus rules as previously, completed with the following one :

$$\boxed{+} \cdot \boxed{\pm} = \boxed{-} \cdot \boxed{\pm} = \boxed{?}.$$

If there exists such an expression of $\det(M)$ by expansion whose formal sign is $\boxed{+}$ or $\boxed{-}$, then \mathcal{C} is called *formally fixed by expansion*.

formally fixed by expansion

Observation 3.5. *If \mathcal{C} is formally fixed by expansion, then \mathcal{C} is fixed.*

The above observation is similar to Observation 3.4 : if \mathcal{C} is formally fixed by expansion then $\sigma_{\mathcal{C}}(\det(M))$ is given as the formal sign of any expression certifying that \mathcal{C} is formally fixed by expansion. Notice that if \mathcal{C} is formally fixed by expansion then all those configurations \mathcal{C}_k induced by \mathcal{C} are fixed, since one must have $\sigma_{\mathcal{C}_k}(\det(M_{\mathcal{E}\setminus\{e_i\}, \mathcal{B}\setminus\{b_k\}})) \in \{\boxed{+}, \boxed{-}\}$.

Conjecture 2. *Let \mathcal{C} be a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$. Then \mathcal{C} is fixed if and only if \mathcal{C} is formally fixed by expansion.*

We point out that if Conjecture 1 is true in dimension $n - 1$, then Conjecture 2 in dimension n implies Conjecture 1 in dimension n . Indeed, in this case, the fixity of the $(n - 1)$ -dimensional configurations corresponding to cofactors can be determined using formal expressions.

Remark. When $n = 4$, consider any 3×3 matrix M' obtained from M by subtracting some columns, and deleting the first row and one column, so that every entry in M' is of type $x_{e,b} - x_{f,b}$ for $e, f \in \mathcal{E}$ and $b \in \mathcal{B}$. We have either $\det(M') = \det(M)$ or $\det(M') = -\det(M)$. Then replace in the matrix M' each formal expression $x_{e,b} - x_{f,b}$ with its formal sign $\sigma_{\mathcal{C}}(x_{e,b} - x_{f,b})$ w.r.t. to a given configuration \mathcal{C} . We obtain a 3×3 matrix with entries in $\{\boxed{+}, \boxed{-}\}$. The point of this remark is that computing formally the sign of the determinant of this last matrix, using the same formal rules as above, will always provide the result $\boxed{?}$. The proof of this property is left to the reader. This shows that a formal matrix M' as above cannot be used alone to derive a formal expression of the determinant proving the fixity of a configuration. One would always need to transform submatrices of M' , which is what we do implicitly by the inductive use of formal signs of induced configurations to determine formal fixity.

Finally, the point of this paper is to deal with the property of being formally fixed by expansion as an inductive criterion for fixity. In what follows, we will prove Conjecture 2 for $n = 4$, together with more precise and direct characterizations in this case (see Theorem 3.10).

3.2.4 A non-fixity criterion

The following Lemma 3.6 will be our main tool to prove that a configuration is non-fixed. We point out that, when $n = 4$, the sufficient condition for being non-fixed provided by Lemma 3.6 turns out to be a necessary and sufficient condition (see Theorem 3.11). However, the authors feel that this equivalence result is too hazardous to be stated as a general conjecture in dimension n .

Lemma 3.6. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$. If there exist $e \in \mathcal{E}$ and $b \in \mathcal{B}$ satisfying the following properties :*

- *e is extreme in the ordering $<_b$ of \mathcal{C} and*
- *the configuration \mathcal{C}' induced by \mathcal{C} on $(\mathcal{E} \setminus \{e\}, \mathcal{B} \setminus \{b\})$ is non-fixed,*

then \mathcal{C} is non-fixed.

Preuve. To lighten notations, let us denote $\mathcal{E} = \{1, \dots, n\}$ and $\mathcal{B} = \{1, \dots, n - 1\}$. Up to equivalence of configurations, we can assume that $e = 1$, that $b = 1$ and that 1 is minimal in the ordering $<_1$.

The expression of $\det(M)$ by expansion with respect to $(1, 2)$ yields :

$$\begin{aligned} \det(M_{\mathcal{E},\mathcal{B}}) &= \sum_{k=1}^{n-1} (-1)^k \cdot (x_{1,k} - x_{2,k}) \cdot \det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{k\}}) \\ &= (x_{2,1} - x_{1,1}) \cdot \det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}) + P[x_{i,j}]_{(i,j)\neq(1,1)}. \end{aligned}$$

where $P[x_{i,j}]_{(i,j)\neq(1,1)}$ is a polynomial on the same variables as $M_{\mathcal{E},\mathcal{B}}$ not depending on $x_{1,1}$.

By hypothesis, the configuration \mathcal{C}' is non-fixed, that is $\sigma_{\mathcal{C}'}(\det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}})) = \boxed{\pm}$. By Lemma 3.1, there exist real values \mathcal{P}'_+ and \mathcal{P}'_- for the entries of this matrix, that is two sets of $n-1$ points labeled by $\mathcal{E}\setminus\{1\}$ in dimension $n-2$, such that $\det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}(\mathcal{P}'_+)) > 0$ and $\det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}(\mathcal{P}'_-)) < 0$.

Let us define a set of n points \mathcal{P}_+ labeled by \mathcal{E} in dimension $n-1$ the following way. The formal variables in $M_{\mathcal{E},\mathcal{B}}$ having real values specified by \mathcal{P}'_+ have the same values in \mathcal{P} . All values not specified by \mathcal{P}'_+ except $x_{1,1}$ are fixed arbitrarily but consistently with the orderings in \mathcal{C} . The value $x_{1,1}$ is chosen small enough so that $x_{1,1}$ is minimal in $<_1$ and

$$(x_{2,1} - x_{1,1}) \cdot \det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}(\mathcal{P}'_+)) > -P[x_{i,j}]_{(i,j)\neq(1,1)}$$

This is possible since $\det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}(\mathcal{P}'_+)) > 0$ and the second term of the inequality does not depend on $x_{1,1}$. By this definition, we have got $\det(M(\mathcal{P}_+)) > 0$.

Similarly, we define \mathcal{P}_- by choosing $x_{1,1}$ small enough so that $x_{1,1}$ is minimal in $<_1$ and

$$(x_{2,1} - x_{1,1}) \cdot \det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}(\mathcal{P}'_-)) < -P[x_{i,j}]_{(i,j)\neq(1,1)}$$

This is possible since $\det(M_{\mathcal{E}\setminus\{1\},\mathcal{B}\setminus\{1\}}(\mathcal{P}'_-)) < 0$ and the second term of the inequality does not depend on $x_{1,1}$. By this definition, we have got $\det(M(\mathcal{P}_-)) < 0$.

We have built \mathcal{P}_+ and \mathcal{P}_- providing opposite signs to real evaluations of $\det(M_{\mathcal{E},\mathcal{B}})$. That is, by Lemma 3.1, \mathcal{C} is non-fixed. \square

3.3 Characterizations in low dimensions

3.3.1 Results in dimension 2

In this section we fix $n = 3$ and $\mathcal{E} = \{A, B, C\}$. In order to lighten notations of variables $x_{e,b}$ for $e \in \mathcal{E}$ and $b \in \mathcal{B}$, we rather denote :

$$M = \begin{pmatrix} 1 & 1 & 1 \\ x_A & x_B & x_C \\ y_A & y_B & y_C \end{pmatrix}$$

We will denote also $\mathcal{B} = \{x, y\}$ and $<_x, <_y$ the orderings in a configuration.

It is easy to check that, up to equivalence of configurations, there exist exactly two linear ordering configurations :

$$\begin{array}{ll} A <_x B <_x C & A <_x B <_x C \\ A <_y B <_y C & B <_y C <_y A \end{array}$$

which correspond respectively to the following grid representations :

		C
	B	
A		

A		
		C
	B	

Théorème 3.7. Let \mathcal{C} be a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 3$, $\mathcal{E} = \{A, B, C\}$ and $\mathcal{B} = \{x, y\}$. The following properties are equivalent :

- a) \mathcal{C} is non-fixed ;
- b) the two orderings on \mathcal{E} in \mathcal{C} are either equal or equal to reversions of each other ;
- c) up to equivalence, \mathcal{C} is equal to

$$\begin{aligned} A <_x B <_x C \\ A <_y B <_y C \end{aligned}$$

Preuve. The equivalence between b) and c) is straightforward and left to the reader. Let us prove that c) implies a). Let \mathcal{C} be given by condition c). Let us choose \mathcal{P} satisfying \mathcal{C} and $x_A = y_A$, $x_B = y_B$, $x_C = y_C$. We have $\det(M(\mathcal{P})) = 0$, hence \mathcal{C} is non-fixed by Lemma 3.1. In order to prove that a) implies c), we can equally prove that the other possible linear ordering configuration (up to equivalence) is fixed. This result is given by Theorem 3.8 below. \square

Théorème 3.8. Let \mathcal{C} be a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 3$, $\mathcal{E} = \{A, B, C\}$ and $\mathcal{B} = \{x, y\}$. The following properties are equivalent :

- a) \mathcal{C} is fixed ;
- b) \mathcal{C} is formally fixed ;
- c) up to equivalence, \mathcal{C} is equal to

$$\begin{aligned} A <_x B <_x C \\ B <_y C <_y A \end{aligned}$$

Preuve. Recall that b) implies a) is always true. Let us prove that c) implies b). We have :

$$\begin{aligned} \det(M) &= \det \begin{pmatrix} 0 & 1 & 0 \\ x_{A-B} & x_B & x_{C-B} \\ y_{A-B} & y_B & y_{C-B} \end{pmatrix} \\ &= (x_A - x_B) \cdot (y_C - y_B) - (y_A - y_B) \cdot (x_C - x_B). \end{aligned}$$

The formal sign of this expression of $\det(M)$ w.r.t. \mathcal{C} is

$$\boxed{-} \cdot \boxed{+} - \boxed{+} \cdot \boxed{+} = \boxed{-}.$$

Hence \mathcal{C} is formally fixed.

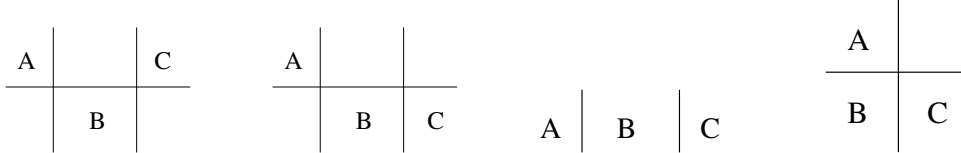
Finally, to prove that a) implies c), we just need to prove that the other linear ordering configuration (up to equivalence) is non-fixed. This has been already proved in the proof of Theorem 3.7. \square

Now that we have listed fixed and non-fixed linear ordering configurations, we are able to determine all fixed and non-fixed configurations using Proposition 3.3. It is easy to check that there are

four ordering configurations which are not linear (up to equivalence of configurations, and omitting those obviously non-fixed for which two elements of \mathcal{E} are comparable in no ordering in the configuration) :

$A <_x B <_x C$ $B <_y A$ $B <_y C$ fixed	$A <_x B <_x C$ $B <_y A$ $C <_y A$ non-fixed (because of $C <_y B <_y A$, and implying the non-fixity of the next ones)	$A <_x B <_x C$ non-fixed	$A <_x C$ $B <_x C$ $B <_y A$ $C <_y A$ non-fixed
--	--	------------------------------	---

Those configurations can be represented respectively in the following grids :



3.3.2 Results in dimension 3

In this section we fix $n = 4$ and $\mathcal{E} = \{A, B, C, D\}$. In order to lighten notations of variables $x_{e,b}$ for $e \in \mathcal{E}$ and $b \in \mathcal{B}$, we rather denote :

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 \\ x_A & x_B & x_C & x_D \\ y_A & y_B & y_C & y_D \\ z_A & z_B & z_C & z_D \end{pmatrix}$$

We will denote also $\mathcal{B} = \{x, y, z\}$ and $<_x, <_y, <_z$ the orderings in a configuration.

As noticed in Section 3.2, in order to prove that a configuration \mathcal{C} is formally fixed by expansion, we need to find an element $e \in E$ such that all the configurations induced by \mathcal{C} on $\mathcal{E} \setminus \{e\}$ are fixed. The proposition below characterizes such induced configurations.

Proposition 3.9. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 4$, $\mathcal{E} = \{A, B, C, D\}$ and $\mathcal{B} = \{x, y, z\}$. All the configurations induced by \mathcal{C} on $\{A, B, C\}$ are fixed if and only if \mathcal{C} is equivalent to a configuration whose orderings satisfy :*

$$\begin{array}{l} B <_x C <_x A \\ C <_y A <_y B \\ A <_z B <_z C \end{array}$$

Preuve. \Rightarrow) The configuration induced by \mathcal{C} on $(\{A, B, C\}, \{y, z\})$ is fixed. According to Theorem 3.8, this implies that the restriction of $<_y$ and $<_z$ to $\{A, B, C\}$ are not equal nor reversions of each other. Similarly, the restriction of $<_x$ and $<_y$, as well as the restriction of $<_x$ and $<_z$, to $\{A, B, C\}$ are not

equal nor reversions of each other. These three properties imply easily the result. We omit the details. \Leftarrow) Direct by Theorem 3.8 using the same criteria as above. \square

Let us now state Theorem 3.10, which is the main theorem of the paper. Its proof is the content of Section 3.3.4. This proof will prove also Theorem 3.11 below at the same time.

Théorème 3.10. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 4$, $\mathcal{E} = \{A, B, C, D\}$ and $\mathcal{B} = \{x, y, z\}$. The following propositions are equivalent :*

- a) \mathcal{C} is fixed ;
- b) \mathcal{C} is formally fixed ;
- c) \mathcal{C} is formally fixed by expansion ;
- d) up to equivalence, \mathcal{C} satisfies :

$$\begin{array}{ccccccc} B & <_x & C & <_x & A & & \\ C & <_y & A & <_y & B & & \\ A & <_z & B & <_z & C & & \end{array}$$

and there exists $X \in \{A, B, C\}$ such that either $X <_b D$ for every $b \in \mathcal{B}$, or $D <_b X$ for every $b \in \mathcal{B}$.

Théorème 3.11. *Let \mathcal{C} be a linear ordering configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 4$. Then \mathcal{C} is non-fixed if and only if conditions of Lemma 3.6 are satisfied, that is : there exist $e \in \mathcal{E}$ and $b \in \mathcal{B}$ such that the configuration \mathcal{C}' induced by \mathcal{C} on $(\mathcal{E} \setminus \{e\}, \mathcal{B} \setminus \{b\})$ is non-fixed and e is extreme in the ordering $<_b$ of \mathcal{C} .*

Given the integer n and the sets \mathcal{E} and \mathcal{B} as previously, there are $(n!)^{n-1}$ linear ordering configurations on $(\mathcal{E}, \mathcal{B})$. We computed the number of linear ordering configurations up to equivalence from $n = 2$ to $n = 6$, yielding the sequence : 1, 2, 21, 5 097 and 71 965 235. We have no general formula. The Chapter 4 explain this program. This integer sequence has been added to The On-Line Encyclopedia of Integer Sequences [22].

We computed the result provided by Theorem 3.10 to list the fixed linear ordering configurations when $n = 4$. Finally, we get that there are exactly 4 fixed configurations among the 21 linear ordering configurations up to equivalence :

$$\begin{array}{ll} B <_x C <_x A <_x D & B <_x C <_x D <_x A \\ C <_y A <_y B <_y D & C <_y A <_y B <_y D \\ A <_z B <_z C <_z D & A <_z B <_z C <_z D \\ \\ B <_x D <_x C <_x A & B <_x C <_x D <_x A \\ C <_y A <_y B <_y D & C <_y D <_y A <_y B \\ A <_z B <_z C <_z D & A <_z B <_z C <_z D \end{array}$$

The interest of the results of this section is to provide a combinatorial characterization as well as an algorithm deciding if a configuration is fixed or not. Also, we point out that our result statements deal with being fixed or not, but not with the exact value $\boxed{+}$ or $\boxed{-}$ of the considered fixed configuration. This sign can be derived easily from the construction stating the fixity. As well, this sign can be obtained by choosing any set of points \mathcal{P} satisfying the configuration and evaluating the sign

of the real number $\det(M(\mathcal{P}))$. Finally, from the list of fixed linear ordering configurations given above, one may compute the list of all fixed (partial) ordering configurations using Proposition 3.3, but we do not give this list here.

3.3.3 Example continued

Let us apply the previous results to some configurations in the 3D model already shown in Section 3.1.2. We recall that ordering configurations are represented on Figures 3.2 and 3.3, with \mathcal{E} any set of four points, and \mathcal{B} corresponding to the three axis $\{x, y, z\}$.

Example 1. *Fixed linear ordering configurations providing a fixed partial ordering configuration : the configuration on $\mathcal{E} = \{2, 5, 8, 9\}$ is fixed.*

This configuration is given by the orderings :

$$\begin{aligned} 9 <_x 5 <_x 2 <_x 8 \\ 5 <_y 8 <_y 9 <_y 2 \\ 2 <_z 8 <_z 9 \quad \text{and} \quad 5 <_z 8 <_z 9 \end{aligned}$$

Its two linear extensions, respectively \mathcal{C}_1 and \mathcal{C}_2 , are the following :

$$\begin{array}{ll} 9 <_x 5 <_x 2 <_x 8 & 9 <_x 5 <_x 2 <_x 8 \\ 5 <_y 8 <_y 9 <_y 2 & 5 <_y 8 <_y 9 <_y 2 \\ 2 <_z 5 <_z 8 <_z 9 & 5 <_z 2 <_z 8 <_z 9 \end{array}$$

Let us write these orderings in another way :

$$\begin{array}{ll} 2 <_z 5 <_z 8 <_z 9 & 5 <_z 2 <_z 8 <_z 9 \\ 5 <_y 8 <_y 9 <_y 2 & 5 <_y 8 <_y 9 <_y 2 \\ 9 <_x 5 <_x 2 <_x 8 & 9 <_x 5 <_x 2 <_x 8 \end{array}$$

By this way, we see that, up to a permutation of \mathcal{B} , that is for $\{i, j, k\} = \{x, y, z\}$, and if we choose $A = 9$, $B = 2$, $C = 8$ and $D = 5$, then the orderings in those configurations both satisfy :

$$\begin{aligned} B <_i C <_i A \\ C <_j A <_j B \\ A <_k B <_k C \end{aligned}$$

as required by Theorem 3.10. Moreover, for each of these orderings we have D smaller than C (i.e. $5 <_x 8$, $5 <_y 8$, $5 <_z 8$). So, by Theorem 3.10, those two configurations are fixed. Then, \mathcal{C} is fixed by Proposition 3.3.

Example 2. *A non-fixed ordering configuration implied by a non-fixed linear ordering configuration : the configuration on $\mathcal{E} = \{1, 3, 7, 10\}$ is non-fixed.*

It is given by the orderings :

$$\begin{aligned} 7 <_x 3 <_x 10 \quad \text{and} \quad 7 <_x 1 <_x 10 \\ 7 <_y 3 <_y 1 \quad \text{and} \quad 7 <_y 10 <_y 1 \\ 1 <_z 7 <_z 10 \quad \text{and} \quad 3 <_z 7 <_z 10 \end{aligned}$$

One of its linear extensions is \mathcal{C}' :

$$\begin{aligned} 7 <_x 3 <_x 1 <_x 10 \\ 7 <_y 10 <_y 3 <_y 1 \\ 3 <_z 1 <_z 7 <_z 10 \end{aligned}$$

The configuration induced by \mathcal{C}' on $(\{7, 3, 1\}, \{x, y\})$ is

$$\begin{aligned} 7 <_x 3 <_x 1 \\ 7 <_y 3 <_y 1 \end{aligned}$$

which is non-fixed by Theorem 3.7. Then 10 is extreme in the ordering $<_z$ of configuration \mathcal{C}' , hence \mathcal{C}' is non-fixed by Lemma 3.6, and so is \mathcal{C} by Proposition 3.3.

Example 3. We leave as an exercise to check using Proposition 3.3 and Theorem 3.10 that the configuration on $\{1, 5, 9, 10\}$ from Section 3.1.2 is fixed. There are four linear extensions to consider, up to symmetries.

Let us conclude by considering completely this example of Section 3.1.2 with ten points in \mathbb{R}^3 . Since there is one configuration for each set of 4 points, there are $\binom{4}{10} = 210$ configurations to study. We wrote a program to test the fixity of these configurations. For each configuration \mathcal{C} the program lists all the linear extensions of \mathcal{C} and computes if each linear extension is fixed or not, based on the results given in Section 3.3.2. Then Proposition 3.3 allows us to conclude. Finally, we find 20 fixed configurations among the 210 configurations.

3.3.4 Proofs of Theorem 3.10 and Theorem 3.11

To prove Theorem 3.10, we study separately : first, the configurations on $(\mathcal{E}, \mathcal{B})$ with $n = 4$ for which there exists a triplet of points $\mathcal{E}' \subseteq \mathcal{E}$ such that all the configurations induced by \mathcal{C} on \mathcal{E}' are fixed (characterized by Proposition 3.9) ; and secondly, the other configurations. The fixed configurations in the first case will be identified. Then every other configuration in the first case and every configuration in the second case will be proved to be non-fixed, using always Lemma 3.6. Hence, Theorem 3.11 will be proved in the meantime.

If all the configurations induced on some triplet are fixed

One may recall that Proposition 3.9 characterizes configurations for which all the configurations induced on some given triplet are fixed.

Proposition 3.12. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 4$, $\mathcal{E} = \{A, B, C, D\}$ and $B = \{x, y, z\}$ such that :*

$$\begin{aligned} B <_x C <_x A \\ C <_y A <_y B \\ A <_z B <_z C \end{aligned}$$

Then \mathcal{C} is formally fixed by expansion if and only if there exists $X \in \{A, B, C\}$ such that either $X <_b D$ for every $b \in \mathcal{B}$, or $D <_b X$ for every $b \in \mathcal{B}$.

Preuve. \Leftarrow) Let us denote $\mathcal{E}' = \{A, B, C\}$ and \mathcal{C}_b the configuration induced by \mathcal{C} on $(\mathcal{E}', \mathcal{B} \setminus \{b\})$ for $b \in \mathcal{B}$. The expansion of $\det(M)$ w.r.t. (D, X) yields :

$$\det(M) = x_{D-X} \cdot \det(M_{\mathcal{E}', \mathcal{B} \setminus \{x\}}) - y_{D-X} \cdot \det(M_{\mathcal{E}', \mathcal{B} \setminus \{y\}}) + z_{D-X} \cdot \det(M_{\mathcal{E}', \mathcal{B} \setminus \{z\}}).$$

We have

$$\begin{aligned} \det(M_{\mathcal{E}', \mathcal{B} \setminus \{x\}}) &= \det \begin{pmatrix} 1 & 1 & 1 \\ y_A & y_B & y_C \\ z_A & z_B & z_C \end{pmatrix} \\ &= \det \begin{pmatrix} y_{B-A} & y_{C-A} \\ z_{B-A} & z_{C-A} \end{pmatrix} \\ &= (y_B - y_A) \cdot (z_C - z_A) - (z_B - z_A) \cdot (y_C - y_A). \end{aligned}$$

whose formal sign w.r.t. \mathcal{C}_x is :

$$(\boxed{+} \cdot \boxed{+}) - (\boxed{+} \cdot \boxed{-}) = \boxed{+}.$$

Similarly, we have

$$\det(M_{\mathcal{E}', \mathcal{B} \setminus \{z\}}) = \det \begin{pmatrix} x_{B-A} & x_{C-A} \\ y_{B-A} & y_{C-A} \end{pmatrix} = (x_B - x_A) \cdot (y_C - y_A) - (y_B - y_A) \cdot (x_C - x_A)$$

whose formal sign w.r.t. \mathcal{C}_z is

$$(\boxed{-} \cdot \boxed{-}) - (\boxed{+} \cdot \boxed{-}) = \boxed{+}.$$

And we have

$$\det(M_{\mathcal{E}', \mathcal{B} \setminus \{y\}}) = \det \begin{pmatrix} x_{B-A} & x_{C-B} \\ z_{B-A} & z_{C-B} \end{pmatrix} = (x_B - x_A) \cdot (z_C - z_B) - (z_B - z_A) \cdot (x_C - x_B)$$

whose formal sign w.r.t. \mathcal{C}_y is

$$(\boxed{-} \cdot \boxed{+}) - (\boxed{+} \cdot \boxed{+}) = \boxed{-}.$$

Now, if the formal signs of x_{D-X} , y_{D-X} and z_{D-X} are all positive (resp. negative), then the formal sign of the above expression of $\det(M)$ w.r.t. \mathcal{C} is $\boxed{+}$ (respectively $\boxed{-}$), which proves that \mathcal{C} is formally fixed by expansion.

\Rightarrow) We will prove the contrapositive : we assume that there exist no $X \in \{A, B, C\}$ such that $X <_b D$ for all $b \in \mathcal{B}$ or such that $D <_b X$ for all $b \in \mathcal{B}$, and we want to prove that \mathcal{C} is non-fixed. Equivalently, we assume that, for every $X \in \{A, B, C\}$, there exist two orderings in \mathcal{C} such that X is smaller than D in an ordering and D is smaller than X in the other ordering. Let us consider two cases. Observe that we will always use Lemma 3.6 to prove that \mathcal{C} is non-fixed.

Case 1 .: *there exist two orderings $<_i$ and $<_j$ in \mathcal{C} , for $i, j \in \mathcal{B}$, such that, for every $X \in \{A, B, C\}$, we have either $X <_i D$ and $D <_j X$, or $X <_j D$ and $D <_i X$. With the restrictions of \mathcal{C} to $\{A, B, C\}$ given in the hypothesis of the proposition, it is easy to check that only 3 configurations satisfy this assumption, up to equivalence (i.e. up to some permutations of \mathcal{B} and \mathcal{E} , and up to ordering reversions).*

$$\begin{array}{cccccc}
 B & <_x & D & <_x & C & <_x & A \\
 C & <_y & A & <_y & D & <_y & B \\
 A & <_z & B & <_z & C & & \\
 \\
 D & <_x & B & <_x & C & <_x & A \\
 C & <_y & A & <_y & B & & \\
 A & <_z & B & <_z & C & <_z & D \\
 \\
 D & <_x & B & <_x & C & <_x & A \\
 C & <_y & A & <_y & B & & \\
 A & <_z & B & <_z & C & <_z & C
 \end{array}$$

We denote these configurations \mathcal{C}_1 , \mathcal{C}_2 and \mathcal{C}_3 . In these configurations, one ordering is a partial ordering : each of these configurations represents four linear ordering configurations.

Let us prove that \mathcal{C}_1 is non-fixed. First, assume that $A <_z D$. The configuration induced by \mathcal{C}_1 on $(\{B, C, D\}, \{x, y\})$ is :

$$\begin{array}{cccc}
 B & <_x & D & <_x & C \\
 C & <_y & D & <_y & B
 \end{array}$$

Since this configuration is non-fixed by Theorem 3.7, and since A is minimal in the ordering $<_z$, then \mathcal{C}_1 is non-fixed by Lemma 3.6. Secondly, and similarly, if $D <_z A$ then the configuration induced by \mathcal{C}_1 on $(\{A, C, D\}, \{y, z\})$ is non-fixed by Theorem 3.7 :

$$\begin{array}{cccc}
 C & <_y & A & <_y & D \\
 D & <_z & A & <_z & C
 \end{array}$$

And B is minimal in $<_x$, so \mathcal{C}_1 is non-fixed by Lemma 3.6.

Now we will show that \mathcal{C}_2 is non-fixed. The proof is similar. Assume that $A <_z D$. As before, A is minimal in the ordering $<_z$. The configuration induced by \mathcal{C}_2 on $(\{B, C, D\}, \{x, y\})$ is non-fixed, so \mathcal{C}_2 is non-fixed by Lemma 3.6. Assume that $D <_z A$. Then the configuration induced by \mathcal{C}_2 on $(\{A, B, D\}, \{x, y\})$ is non-fixed. Since C is maximal in the ordering $<_z$, we get that \mathcal{C}_2 is non-fixed by Lemma 3.6.

We will use the same method to prove that \mathcal{C}_3 is non-fixed. First, assume that $C <_y D$. The configuration induced by \mathcal{C}_3 on $(\{A, B, D\}, \{x, z\})$ is non-fixed. Since C is minimal in the ordering $<_y$, then \mathcal{C}_3 is non-fixed by Lemma 3.6. Secondly, if $D <_y C$ then the configuration induced by \mathcal{C}_3 on $(\{B, C, D\}, \{y, z\})$ is non-fixed. Since A is maximal in the ordering $<_x$, then \mathcal{C}_3 is non-fixed by Lemma 3.6.

Case 2 .: *the assumption of case 1 does not hold.* It is easy to check that, up to equivalence of configurations, there are two other configurations such that X is smaller than D in an ordering and D is smaller than X in another ordering :

$$\begin{array}{cccccc}
 D & <_x & B & <_x & C & <_x & A \\
 C & <_y & A & <_y & D & <_y & B \\
 A & <_z & B & <_z & D & <_z & C \\
 \\
 B & <_x & D & <_x & C & <_x & A \\
 C & <_y & D & <_y & A & <_y & B \\
 A & <_z & D & <_z & B & <_z & C
 \end{array}$$

We denote these configurations \mathcal{C}_4 and \mathcal{C}_5 .

We first prove that \mathcal{C}_4 is non-fixed. The configuration induced by \mathcal{C}_4 on $(\{B, C, D\}, \{y, z\})$ is non-fixed (again by Theorem 3.7), and A is maximal in the ordering $<_x$. So the configuration \mathcal{C}_4 is non-fixed by Lemma 3.6.

Similarly, the configuration induced by \mathcal{C}_5 on $(\{B, C, D\}, \{x, y\})$ is non-fixed. Since A is minimal in the ordering $<_z$, \mathcal{C}_5 is non-fixed by Lemma 3.6. \square

If for every triplet there is at least one non-fixed induced configuration

We call *triplet* a set having three elements. For short, we may denote ABC the triplet $\{A, B, C\}$. Consider two linear orderings $<_i$ and $<_j$ on a same set containing the triplet $\{A, B, C\}$ and assume, without loss of generality, that $A <_i B <_i C$. We say that $<_i$ and $<_j$ are *equal*, resp. *reversed*, on the triplet $\{A, B, C\}$, if $A <_j B <_j C$, resp. $C <_j B <_j A$. Recall during the whole section that, by Theorem 3.7, $<_i$ and $<_j$ are equal or reversed on $\{A, B, C\}$ if and only if the configuration formed by $<_i$ and $<_j$ on $(\{A, B, C\}, \{i, j\})$ is non-fixed. In what follows, we will often consider two such orderings equal or reversed on such a triplet. For short, we may say also that the triplet $\{A, B, C\}$ is *conformal* w.r.t. the two orderings $<_i$ and $<_j$.

triplet

equal
reversed

conformal

Lemma 3.13. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 4$, $\mathcal{E} = \{A, B, C, D\}$ and $\mathcal{B} = \{x, y, z\}$. Assume that $A <_x B <_x C <_x D$. Consider the three pairs of triplets $\{\{A, B, C\}, \{B, C, D\}\}$, $\{\{A, B, C\}, \{A, C, D\}\}$ and $\{\{A, B, D\}, \{B, C, D\}\}$. If, for at least one of these pairs, the two orderings $<_x$ and $<_y$ are equal or reversed on both the two triplets of the pair, then $<_x$ and $<_y$ are equal or reversed on $\{A, B, C, D\}$, that is: either $A <_y B <_y C <_y D$ or $D <_y C <_y B <_y A$.*

Preuve. First, we prove that if two orderings $<_i$ and $<_j$ are equal or reversed on two triplets of \mathcal{E} , then they are either equal on the two triplets, or reversed on the two triplets. These two triplets have two elements in common. We denote these triplets XYZ and XYW . Up to relabelling, we can assume that $X <_i Y$. Assume for a contradiction that $<_i$ and $<_j$ are equal on XYZ and reversed on XYW . The triplet XYZ shows that $X <_j Y$, whereas XYW shows that $Y <_j X$, which is a contradiction.

Back to the pairs of triplets $\{ABC, BCD\}$, $\{ABC, ACD\}$ and $\{ABD, BCD\}$. Let us first consider the pair $\{ABC, BCD\}$. Assume that $<_x$ and $<_y$ are equal on ABC and BCD . We prove that $<_x$ and $<_y$ are equal on \mathcal{E} . We have $A <_x B <_x C$ by hypothesis of the lemma, and $<_x$ and $<_y$ equal on ABC by assumption, so we have $A <_y B <_y C$. We have $C <_x D$ by hypothesis, and $<_x$ and $<_y$ equal on BCD by assumption, so we have $C <_y D$. Hence, the ordering $<_y$ is $A <_y B <_y C <_y D$. If we consider the pair $\{ABC, ACD\}$, then we obtain the same result with the same proof. Consider the pair $\{ABD, BCD\}$, and assume that $<_x$ and $<_y$ are equal on ABD and BCD . Then the ordering $<_y$ restricted to ABD is $A <_y B <_y D$. The orderings $<_x$ and $<_y$ are also equal on BCD . So we have $B <_y C <_y D$. This proves that we have $A <_y B <_y C <_y D$.

Now, assume that there is a pair of triplets such that $<_x$ and $<_y$ are reversed on the two triplets of the pair. Let us denote $<_{opp(y)}$ the reversion of the ordering $<_y$. Then $<_x$ and $<_{opp(y)}$ are equal on the two triplets of the pair. From the result above, we get $A <_{opp(y)} B <_{opp(y)} C <_{opp(y)} D$, that is $D <_y C <_y B <_y A$. \square

Proposition 3.14. *Let \mathcal{C} be a configuration on $(\mathcal{E}, \mathcal{B})$ with $n = 4$. If for every triplet $\mathcal{E}' \subseteq \mathcal{E}$ there exists $b \in \mathcal{B}$ such that the configuration induced by \mathcal{C} on $(\mathcal{E}', \mathcal{B} \setminus \{b\})$ is non-fixed, then \mathcal{C} is non-fixed.*

Preuve. We consider separate cases. To this aim, for each ordering $<_b$ in \mathcal{C} , we count the number of non-fixed configurations induced by \mathcal{C} on $(\mathcal{E}', \mathcal{B} \setminus \{b\})$ with $\mathcal{E}' = \mathcal{E} \setminus \{e\}$ for some $e \in \mathcal{E}$. By Theorem 3.7, it amounts to count, for each pair of orderings in \mathcal{C} (i.e. for each $\mathcal{B} \setminus \{b\}$), the number of triplets \mathcal{E}' in $\{ABC, ABD, ACD, BCD\}$ which are conformal w.r.t. this pair of orderings. Without loss of generality (up to a permutation of \mathcal{B}), we can assume that the pair of orderings $\{<_x, <_y\}$ maximizes this number of triplets. Also, without loss of generality (up to a permutation of \mathcal{E}), we can assume that $A <_x B <_x C <_x D$. The different cases correspond to the number of triplets of \mathcal{E} conformal w.r.t. $<_x$ and $<_y$.

Case 1 : *four triplets of \mathcal{E} are conformal w.r.t. $<_x$ and $<_y$.*

All the triplets of \mathcal{E} are conformal w.r.t. $<_x$ and $<_y$. So we have $A <_y B <_y C <_y D$ or $D <_y C <_y B <_y A$ by Lemma 3.13. Let $X \in \mathcal{E}$ be an extreme element in the ordering $<_z$. The triplet $\mathcal{E}' = \{A, B, C, D\} \setminus \{X\}$ is conformal w.r.t. $<_x$ and $<_y$. That is, by Theorem 3.7 : the configuration induced by \mathcal{C} on $(\mathcal{E}', \{x, y\})$ is non-fixed. Hence \mathcal{C} is non-fixed by Lemma 3.6.

Case 2 : *three triplets of \mathcal{E} are conformal w.r.t. $<_x$ and $<_y$.*

We prove that this case is not possible. Otherwise, the set of three triplets contains one of the pairs $\{ABC, BCD\}$, $\{ABC, ACD\}$, $\{ABD, BCD\}$. Then we have $A <_y B <_y C <_y D$ or $D <_y C <_y B <_y A$ by Lemma 3.13. So the four triplets are conformal w.r.t. $<_x$ and $<_y$.

Case 3 : *two triplets of \mathcal{E} are conformal w.r.t. $<_x$ and $<_y$.*

This is the more involved case. First, we build a bipartite graph G between the four triplets of \mathcal{E} and the three orderings of \mathcal{C} . Let us denote T_1, T_2, T_3, T_4 those triplets, and O_1, O_2, O_3 those orderings (we will study later in our case analysis which label corresponds to which triplet/ordering). Thus, the edges are of the form (T, O) with T in $\{T_1, \dots, T_4\}$ and O in $\{O_1, O_2, O_3\}$. The graph G is defined the following way : if a triplet T is conformal w.r.t. the two orderings O_a and O_b , for $a, b \in \{1, 2, 3\}$, then both (T, O_a) and (T, O_b) are edges of G ; and every edge (T, O_a) in G means that there exists O_b such that T is conformal w.r.t. O_a and O_b .

Let us state several useful Claims.

1. *We have : (T, O_a) and (T, O_b) are edges of G if and only if T is conformal w.r.t. the two orderings O_a and O_b .* Indeed, if (T, O_a) and (T, O_b) are edges, but T is not conformal w.r.t. the two orderings O_a and O_b , then there exist an edge (T, O_c) in G , with T conformal w.r.t. the two orderings O_a and O_c , and T conformal w.r.t. O_b and O_c , which implies directly that the orderings O_a and O_b are equal or reversed on T , that is T is conformal w.r.t. O_a and O_b .
2. *We have : the two vertices $<_x$ and $<_y$ of G have exactly two neighbors in common.* Indeed, exactly two triplets are conformal w.r.t. $<_x$ and $<_y$ by assumption.
3. *We have : if $<_x$ and another vertex in $\{O_1, O_2, O_3\}$ are adjacent in G to the same two vertices in $\{T_1, \dots, T_4\}$, then these two vertices in $\{T_1, \dots, T_4\}$ form one of these pairs : $\{ABC, ABD\}$, or $\{ABD, ACD\}$, or $\{ACD, BCD\}$.* Indeed, if two triplets of \mathcal{E} are conformal w.r.t. the same two orderings $<_x$ and $<_a$ in \mathcal{C} , for $a \in \{y, z\}$, then these two triplets are not $\{ABC, BCD\}$, nor $\{ABC, ACD\}$, nor $\{ABD, BCD\}$: otherwise, by Lemma 3.13, the four triplets of \mathcal{E} would be

conformal w.r.t. the two orderings, a contradiction with the maximality property of $<_x$ and $<_y$ and the assumption of Case 3.

4. *We have : in the graph G , each vertex in $\{T_1, \dots, T_4\}$ is not isolated, and is adjacent to at least two vertices in $\{O_1, O_2, O_3\}$.* Indeed, by hypothesis of the proposition, for every triplet T of \mathcal{E} , there exists $b \in \mathcal{B}$ such that the configuration induced by \mathcal{C} on $(T, \mathcal{B} \setminus \{b\})$ is non fixed, that is, Theorem 3.7, such that T is conformal w.r.t. the two orderings in \mathcal{C} different from $<_b$.
5. *We have : there are at least eight edges in G .* This is direct by Claim 4 above.
6. *We have : all the vertices in $\{O_1, O_2, O_3\}$ have degree at least two.* Indeed, assume that a vertex in $\{O_1, O_2, O_3\}$ has degree zero or one, then, among the two other vertices in $\{O_1, O_2, O_3\}$, one has degree four and the other has degree at least three. Hence, those two orderings are adjacent to three common triplets. We get a contradiction with the definition of $<_x$ and $<_y$ as maximizing the number of triplets adjacent to both two given orderings (we assumed that this number is equal to two).

By Claim 5 and Claim 6, we have that : either there is at least one vertex of G in $\{O_1, O_2, O_3\}$ with degree 4, or there are at least two vertices of G in $\{O_1, O_2, O_3\}$ with degree 3. Figures 3.7 and 3.8 show respectively the two bipartite graphs with a minimal number of edges and satisfying one of these two properties, up to permutations of $\{T_1, \dots, T_4\}$ and $\{O_1, O_2, O_3\}$. So we now assume, without loss of generality, that removing edges from G leads to one the two graphs depicted on Figures 3.7 and 3.8.

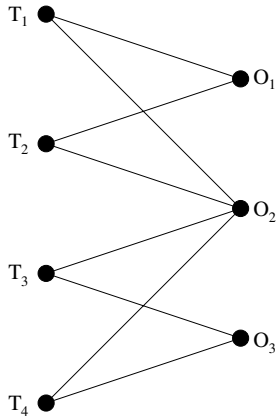


FIGURE 3.7 – G minimal with one vertex having degree 4

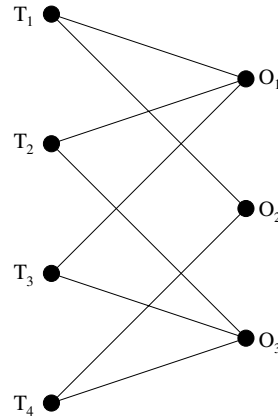


FIGURE 3.8 – G minimal with two vertices having degree 3

Let us consider separately the two cases depending on these two properties.

- G has at least one vertex with degree 4 among $\{O_1, O_2, O_3\}$. See Figure 3.7.

First, assume that the ordering $<_x$ is the vertex O_2 with degree four. In this case, the two triplets T_1 and T_2 are adjacent to the same two orderings O_1 and $<_x$. By Claim 3 above, this implies that T_1 and T_2 form one of these pairs : $\{ABC, ABD\}$, $\{ABD, ACD\}$, $\{ACD, BCD\}$. The same result holds for the two triplets T_3 and T_4 adjacent to O_3 and $<_x$. Since these four triplets are distinct, $\{T_1, T_2\}$ and $\{T_3, T_4\}$ are not equal to $\{ABD, ACD\}$. Without loss of generality, we can assume that $\{T_1, T_2\} = \{ABC, ABD\}$ and $\{T_3, T_4\} = \{ACD, BCD\}$.

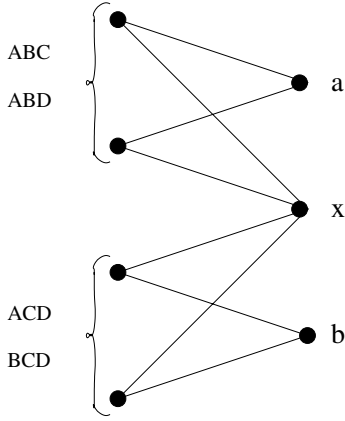


FIGURE 3.9 – O_2 is $<_x$

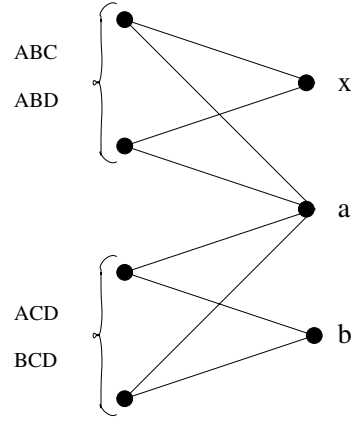


FIGURE 3.10 – O_1 is $<_x$

Let us denote $a, b \in \{y, z\}$ such that $<_a$ is O_1 , and $<_b$ is O_3 . This case is illustrated in Figure 3.9. In this case, ABC is adjacent in G to $<_x$ and $<_a$. Hence, by Claim 1, the configuration induced by \mathcal{C} on $(ABC, \{x, a\})$ is non-fixed. On the other hand, ACD is adjacent to $<_x$ and $<_b$ in G , meaning that $<_x$ and $<_b$ are equal or reversed on ACD . Since D is extreme in $<_x$, then D is extreme in $<_b$ restricted to ACD . Similarly, BCD is adjacent to $<_x$ and $<_b$ in G implies that D is extreme in $<_b$ restricted to BCD . Hence D is extreme in $<_b$. So we get that \mathcal{C} is non-fixed by Lemma 3.6.

Secondly, assume that the ordering $<_x$ is the vertex O_1 or O_3 . Without loss of generality, since O_1 and O_3 play symmetric roles in G , we can assume that $<_x$ is O_1 . By Claim 3, this implies that T_1 and T_2 form one of these pairs : $\{ABC, ABD\}$, $\{ABD, ACD\}$, $\{ACD, BCD\}$.

We will prove that $\{T_1, T_2\}$ is not equal to $\{ABD, ACD\}$. Assume for a contradiction that $\{T_1, T_2\}$ is equal to $\{ABD, ACD\}$. Let $a, b \in \{y, z\}$ such that O_2 is $<_a$ and O_3 is $<_b$. Then, $(ABD, <_x)$ and $(ABD, <_a)$ are edges of G . By Claim 1, ABD is conformal w.r.t. $<_x$ and $<_a$. Since the ordering $<_x$ restricted to ABD is $A <_x B <_x D$, the ordering $<_a$ restricted to ABD is either $A <_a B <_a D$ or $D <_a B <_a A$. Similarly, with the triplet ACD , we have either $A <_a C <_a D$ or $D <_a C <_a A$. If $A <_a B <_a C <_a D$ or $D <_a C <_a B <_a A$, then the four triplets are conformal w.r.t. $<_x$ and $<_a$, which is a contradiction with the choice of $<_x$ and $<_y$ as maximizing the number of triplets conformal with a pair of orderings, and the assumption of Case 3 that this maximal number equals two. So we have either $A <_a C <_a B <_a D$ or $D <_a B <_a C <_a A$. On the other hand, the pair $\{T_3, T_4\}$ is equal to $\{ABC, BCD\}$. Since ABC is conformal w.r.t. $<_a$ and $<_b$, the ordering $<_b$ restricted to ABC is either $A <_b C <_b B$ or $B <_b C <_b A$. Similarly, with the triplet BCD we have either $C <_b B <_b D$ or $D <_b B <_b C$. So we have either $A <_b C <_b B <_b D$ or $D <_b B <_b C <_b A$. Hence the four triplets are conformal w.r.t. $<_a$ and $<_b$, which is, as above, a contradiction with the maximality property of $<_x$ and $<_y$. So $\{T_1, T_2\}$ is not equal to $\{ABD, ACD\}$.

If $\{T_1, T_2\}$ is equal to $\{ABC, ABD\}$. The Figure 3.10 shows this case. We have that BCD is conformal w.r.t. $<_y$ and $<_z$, meaning that the configuration induced by \mathcal{C} on $(BCD, \{y, z\})$ is non-fixed. Since we have $A <_x B <_x C <_x D$, A is minimal in $<_x$. Hence, the configuration \mathcal{C} is non-fixed by Lemma 3.6. If $\{T_1, T_2\}$ is equal to $\{ACD, BCD\}$. We have that ABC is conformal w.r.t. $<_y$ and $<_z$, meaning that the configuration induced by \mathcal{C} on $(ABC, \{y, z\})$ is non-fixed.

Since we have $A <_x B <_x C <_x D$, D is maximal in $<_x$. Hence, the configuration \mathcal{C} is non-fixed by Lemma 3.6.

- The graph G has at least 2 vertices with degree 3 among $\{O_1, O_2, O_3\}$. See Figure 3.8.

By Claim 2, $<_x$ and $<_y$ have exactly two common neighbors. We first prove that, without loss of generality, we can assume that O_1 is $<_x$, and O_3 is $<_y$. First, assume that O_2 is $<_x$ or $<_y$ and O_2 has degree two. Then either O_1 or O_3 has degree four (in order to have two common neighbors with O_2), implying that O_1 and O_3 have three common neighbors, which is a contradiction with the maximality property of $<_x$ and $<_y$ and the assumption of Case 3. So, if O_2 has degree two, then we necessarily have $\{<_x, <_y\} = \{O_1, O_3\}$. Now, if O_2 has degree at least three, then we can exchange O_2 with O_1 or O_3 (and some triplets accordingly) in order to have $\{<_x, <_y\} = \{O_1, O_3\}$. Finally, we have $\{<_x, <_y\} = \{O_1, O_3\}$, and O_1 and O_3 play symmetric roles in G , so we can choose to have $O_1 = <_x$ and $O_3 = <_y$.

By Claim 3, there are 2 cases : either the vertices T_2 and T_3 form the pair $\{ACD, BCD\}$, or they form a pair among $\{ABC, ABD\}$ and $\{ABD, ACD\}$.

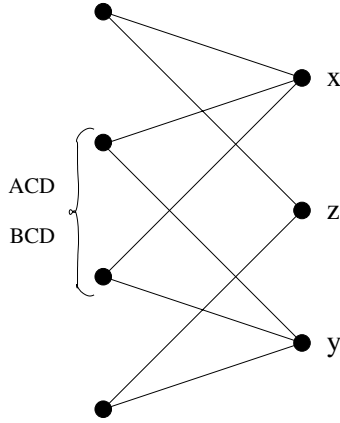


FIGURE 3.11 – $\{T_2, T_3\} = \{ACD, BCD\}$

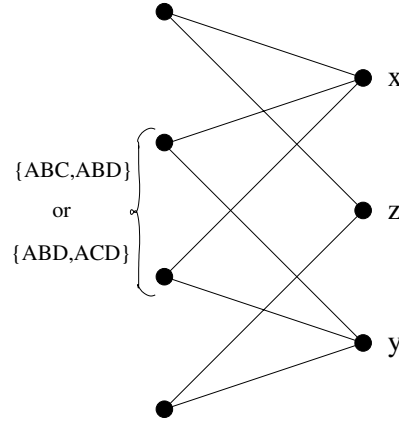


FIGURE 3.12 – $\{T_2, T_3\} \neq \{ACD, BCD\}$

In the first case, we have that ACD and BCD are conformal w.r.t. $<_x$ and $<_y$. Figure 3.11 shows this case. Hence, ABC is conformal w.r.t. $<_z$ and a second ordering $<_a$ for $\{a, b\} = \{x, y\}$. So the configuration induced by \mathcal{C} on $(ABC, \{z, a\})$ is non-fixed, by Theorem 3.7. Assume $b = x$. Since D is extreme in $<_x$, then \mathcal{C} is non-fixed by Lemma 3.6. Now, assume that $b = y$. Since D is extremal in $<_x$, and ACD is conformal w.r.t. $<_x$ and $<_y$, then D is also extreme in $<_y$ restricted to ACD . Similarly, with the triplet BCD , we get that D is extreme in the ordering $<_y$ restricted to BCD and hence extreme in $<_y$. So \mathcal{C} is non-fixed by Lemma 3.6.

In the second case, the pair of triplets conformal w.r.t. $<_x$ and $<_y$ is either $\{ABC, ABD\}$ or $\{ABD, ACD\}$. Figure 3.12 illustrates this case. So BCD is conformal w.r.t. $<_z$ and a second ordering $<_a$, for $\{a, b\} = \{x, y\}$. The configuration induced by \mathcal{C} on $(BCD, \{z, a\})$ is non-fixed, by Theorem 3.7. Assume $b = x$. Since A is extreme in $<_x$, \mathcal{C} is non-fixed by Lemma 3.6. Now, assume $b = y$. Similarly as previously, since A is extreme in $<_x$, we get that A is extreme in $<_y$ when $\{ABC, ABD\}$ are both conformal w.r.t. $<_x$ and $<_y$, as well as when $\{ABD, ACD\}$ are both conformal w.r.t. $<_x$ and $<_y$. So \mathcal{C} is non-fixed by Lemma 3.6.

Case 4 : *zero or one triplets of \mathcal{E} are conformal w.r.t. $<_x$ and $<_y$.* We will prove that this case is not possible. Since there is no triplet of \mathcal{E} such that all the configurations induced by \mathcal{C} on this triplet are fixed, each triplet is conformal w.r.t. at least two orderings of \mathcal{C} . There are three pairs of orderings and four triplets, so there is at least one pair of orderings such that two triplets are conformal w.r.t. this pair of orderings. \square

Final proofs

Proof of Theorem 3.10. Recall that b) implies a) is given by Observation 3.4. We have c) implies b) as remarked below Conjecture 2 : if a configuration induced on a triplet is fixed then it is formally fixed by Theorem 3.8, hence formal fixity by expansion for $n = 4$ implies formal fixity for $n = 4$. We have d) implies c) by Proposition 3.12. At last, to prove that a) implies d), assume d) is false. If \mathcal{C} is equivalent to a configuration whose orderings satisfy

$$\begin{array}{ccccccc} B & <_x & C & <_x & A & & \\ C & <_y & A & <_y & B & & \\ A & <_z & B & <_z & C & & \end{array}$$

then it is non-fixed by Propositions 3.12. If \mathcal{C} is not equivalent to such a configuration, then, by Proposition 3.9 and a permutation of \mathcal{E} , we have that : for every triplet \mathcal{E}' of \mathcal{E} there exists $b \in \mathcal{B}$ such that the configuration induced by \mathcal{C} on $(\mathcal{E}', \mathcal{B} \setminus \{b\})$ is non-fixed. Then \mathcal{C} is non-fixed by Proposition 3.14. So we get that a) is false, meaning that a) implies d). \square

Proof of Theorem 3.11. This proof follows from the proofs of Propositions 3.12 and 3.14 which enumerate, by means of several cases, all possible non-fixed configurations up to equivalence. In every case, the fact that a configuration is non-fixed is proved using Lemma 3.6, up to equivalence. Hence, every non-fixed configuration is equivalent to a configuration satisfying the hypothesis of this lemma. Since, obviously, permutations of \mathcal{B} or \mathcal{E} and ordering reversions do not change this property, we get that every non-fixed configuration satisfies the hypothesis of this lemma. \square

Calcul du nombre d'ensembles d'ordres totaux à transformations près

Comme nous l'avons annoncé dans la Section 3.3.2, nous avons calculé que le nombre de configurations linéaires à équivalence près pour $r = 2$ à $r = 6$ produit la séquence : 1, 2, 21, 5 097 and 71 965 235. Cette séquence a été ajoutée à *The On-Line Encyclopedia of Integer Sequences* [22]. Dans ce chapitre, nous expliquons comment calculer ce nombre en fonction de n .

Dans la Section 3.3.4 nous avons vu que les preuves en dimension trois se basent sur de nombreux cas à étudier. Nous avons donc décidé de calculer le nombre de configurations linéaires à équivalence près en fonction de la dimension. L'idée est que si cette valeur ne grandit pas trop rapidement en fonction de la dimension alors une preuve du même type que celle de la Section 3.3.4 est envisageable en dimension supérieure.

Pour déterminer le nombre de configurations linéaires à équivalence près, nous commençons par générer un sous-ensemble \mathcal{S}_r de l'ensemble de toutes les configurations linéaires. Nous montrons ensuite que chaque classe d'équivalence possède un représentant dans \mathcal{S}_r puis nous décrivons comment éliminer les configurations linéaires qui possèdent déjà dans \mathcal{S}_r une configuration linéaire équivalente. Pour minimiser les calculs il faut donc pouvoir générer un sous-ensemble \mathcal{S}_r le plus petit possible et sa taille va nous fournir une borne supérieure au nombre de configurations linéaires à équivalence près.

4.1 Problématique générale

Nous allons commencer par redonner les définitions du Chapitre 3 qui vont nous servir dans ce chapitre.

Soit \mathcal{E} un ensemble de taille r . On appelle configuration d'ordres sur \mathcal{E} , ou juste configuration, une liste \mathcal{C} de $r - 1$ ordres $\langle_1, \dots, \langle_{r-1}$ sur \mathcal{E} . Dans une configuration, les ordres ne sont pas nécessairement tous différents. On dit qu'une configuration \mathcal{C} est linéaire si tous les ordres \langle_i de \mathcal{C} sont des ordres totaux. On appelle inversion d'un ordre, l'ordre obtenu en inversant toutes les inégalités de cet ordre. Deux configurations sur \mathcal{E} sont dites équivalentes si elles sont égales, à une permutation de \mathcal{E} , une permutation de $\langle_1, \dots, \langle_{r-1}$ et des inversions d'ordres près. Par exemple, les deux

configurations suivantes sont équivalentes :

$$\begin{array}{ll} 1 <_1 2 <_1 3 <_1 4 & 1 <_1 2 <_1 3 <_1 4 \\ 3 <_2 1 <_2 4 <_2 2 & 1 <_2 3 <_2 4 <_2 2 \\ 4 <_3 1 <_3 2 <_3 3 & 4 <_3 1 <_3 2 <_3 3 \end{array}$$

En effet, notons \mathcal{C}_1 et \mathcal{C}_2 ces deux configurations linéaires. On peut alors passer de \mathcal{C}_1 à \mathcal{C}_2 de la façon suivante :

$$\begin{array}{ll} 1 <_1 2 <_1 3 <_1 4 & 3 <_1 2 <_1 1 <_1 4 \\ 3 <_2 1 <_2 4 <_2 2 & \iff 1 <_2 3 <_2 4 <_2 2 \\ 4 <_3 1 <_3 2 <_3 3 & 4 <_3 3 <_3 2 <_3 1 \\ & 4 <_1 1 <_1 2 <_1 3 \\ & \iff 1 <_2 3 <_2 4 <_2 2 \\ & 1 <_3 2 <_3 3 <_3 4 \\ & 1 <_1 2 <_1 3 <_1 4 \\ & \iff 1 <_2 3 <_2 4 <_2 2 \\ & 4 <_3 1 <_3 2 <_3 3 \end{array}$$

Pour obtenir la première équivalence, on applique la permutation de $\mathcal{E} \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 2 & 1 & 4 \end{pmatrix}$ à \mathcal{C}_1 . Dans la deuxième équivalence, on inverse les ordres $<_1$ et $<_3$ et dans la troisième équivalence on permute les ordres $<_1$ et $<_3$.

L'équivalence de \mathcal{C}_1 et \mathcal{C}_2 ne paraissait pas évidente à première vue et une question se pose alors : pour r fixé, quel est le nombre de configurations à équivalence près. Nous noterons $C_{diff}(r)$ ce nombre.

Le problème peut également se présenter de la façon suivante : soit T un tableau de $r - 1$ lignes et r colonnes. Chaque ligne du tableau contient une seule fois tous les entiers de 1 à r . Soient les trois transformations suivantes :

- $f_1(\sigma)$ consiste à appliquer une permutation σ de $\{1, \dots, r\}$ à chaque case du tableau. Formellement $\forall i \in \{0, \dots, r - 2\}, \forall j \in \{0, \dots, r - 1\}, T[i, j] \leftarrow \sigma(T[i, j])$
- $f_2(\pi)$ consiste à appliquer une permutation π de $\{0, \dots, r - 2\}$ aux lignes du tableau. Formellement $\forall i \in \{0, \dots, r - 2\}, \forall j \in \{0, \dots, r - 1\}, T[i, j] \leftarrow T[\pi(i), j]$
- $f_3(i)$ consiste à inverser la ligne i du tableau. Formellement $\forall j \in \{0, \dots, r - 1\}, T[i, j] \leftarrow T[i, (r - 1 - j)]$

Deux tableaux sont dits équivalents s'ils sont égaux à applications de transformations f_1 , f_2 et f_3 près. On cherche alors à déterminer pour r fixé le nombre de tableaux à équivalence près.

Soit l une liste d'éléments de \mathcal{E} telle que tout élément de \mathcal{E} apparaisse exactement une fois dans l . Soit $<_o$ un ordre sur \mathcal{E} . On dit que l est associée à $<_o$ si les éléments de l sont triés selon l'ordre $<_o$. Par exemple la liste $(1, 3, 2, 4)$ est associée à l'ordre $1 < 3 < 2 < 4$.

On représentera chaque ordre linéaire sur \mathcal{E} par un tableau à une dimension dont les cases sont numérotées de 0 à $r - 1$. L'élément le plus petit d'un ordre linéaire est donc dans la case d'indice 0

liste associée à un ordre

indice dans un ordre

du tableau et l'élément le plus grand dans cet ordre linéaire est à la case d'indice $r - 1$. Soit $<_o$ un ordre sur \mathcal{E} . On note $ind(j, <_o)$ l'indice d'un élément j dans $<_o$ ($ind(j, <_o) \in \{0, \dots, r - 1\}$).

Dans la suite de ce chapitre, nous utiliserons des permutations. Nous rappelons alors quelques résultats sur les permutations qui nous seront utiles pour la suite.

Une permutation de $1, \dots, k$ est une bijection de $1, \dots, k$ sur lui-même. En d'autres termes, une bijection est une application qui pour tout élément i de $1, \dots, k$ associe un élément de $1, \dots, k$ de sorte que deux éléments différents ne soient pas associés à un même élément.

Tout comme pour les ordres, on représentera une permutation de $1, \dots, k$ par un tableau de taille k dont les cases sont numérotées de 0 à $k - 1$. Ainsi on dira que l'indice de l'élément j dans la permutation est i (avec $i \in \{0, \dots, k - 1\}$) pour signifier que $i + 1$ est associé à j dans cette permutation.

Le nombre de permutations de $1, \dots, k$ est égal à $k!$. De plus, le nombre de permutations de $1, \dots, k$ telles que l'indice de l'élément i soit j est de $(k - 1)!$. En effet, si l'on voit une permutation comme un tableau, il nous reste $k - 1$ cases du tableau à remplir (les cases d'indice $0, \dots, j - 1, j + 1, \dots, k - 1$) à l'aide de $k - 1$ valeurs ($1, \dots, i - 1, i + 1, \dots, k$). Ainsi le nombre de permutations de $1, \dots, k$ telles que l'indice de l'élément i soit j est égal au nombre de permutations de $1, \dots, k - 1$ soit $(k - 1)!$.

Dans la suite de ce chapitre, nous utiliserons également les ensembles et multi-ensembles. On rappelle que $\binom{s}{k}$ est le nombre de parties à k éléments parmi un ensemble de s éléments. Un *multi-ensemble* est un ensemble dans lequel chaque élément peut apparaître plusieurs fois (contrairement aux ensembles dans lesquels les éléments sont présents au plus une fois). On note alors $\left(\binom{s}{k}\right)$ le nombre de multi-ensembles de k éléments pris parmi un ensemble de s éléments.

indice d'un
élément
dans une
permuta-
tion

multi-
ensemble

Remarque. Comme dans une configuration les ordres ne sont pas nécessairement tous différents, une configuration est donc un multi-ensemble d'ordres sur \mathcal{E} .

4.2 Un encadrement

4.2.1 Une borne supérieure

Nous allons voir comment créer un ensemble de configurations linéaires contenant toutes les configurations linéaires à équivalence près. Pour cela, nous avons introduit trois règles :

Règle 4.1. L'ordre $<_1$ doit être égal à $1 <_1 2 <_1 \dots <_1 r$.

Règle 4.2. En notant l_1, \dots, l_{r-1} les listes associées aux ordres $<_1, \dots, <_{r-1}$ d'une configuration, on doit avoir $l_1 \leq l_2 \leq \dots \leq l_{r-1}$, où \leq est l'ordre lexicographique.

Règle 4.3. Dans chaque ordre $<_i$, la valeur 1 doit faire parti des $\lceil \frac{r}{2} \rceil$ plus petites valeurs. De plus si r est impair et que 1 est la valeur médiane (s'il y a autant d'éléments plus petits que 1 que d'éléments plus grands que 1 dans l'ordre $<_i$) alors la valeur 2 doit être plus petite que 1 dans cet ordre.

Formellement, cette règle impose que pour tout ordre $<_i$ on ait soit $ind(1, <_i) < \lceil \frac{r}{2} \rceil$, soit $ind(1, <_i) = \lceil \frac{r}{2} \rceil$ et $ind(2, <_i) < ind(1, <_i)$.

On note \mathcal{S}_r l'ensemble des configurations linéaires générées par ces trois règles.

\mathcal{S}_r

Pour illustrer ces règles on peut reprendre les configurations linéaires \mathcal{C}_1 et \mathcal{C}_2 utilisées dans la Section 4.1. En effet \mathcal{C}_1 et \mathcal{C}_2 respectent ces trois règles.

Remarque. La Règle 4.1 permet qu'il n'y ait pas deux configurations dans \mathcal{S}_r qui soient égales uniquement à permutation de \mathcal{E} près. La Règle 4.2 permet qu'il n'y ait pas deux configurations dans \mathcal{S}_r qui soient égales à permutation de $\langle_1, \dots, \langle_{r-1}$ près et la Règle 4.3 permet qu'il n'y ait pas deux configurations dans \mathcal{S}_r qui soient égales à inversion d'ordres près. Nous pourrions penser que nous ne pouvons générer uniquement les configurations différentes à équivalence près grâce à ces règles. Malheureusement les configurations \mathcal{C}_1 et \mathcal{C}_2 de la Section 4.1 nous montre le contraire. En effet ces deux configurations respectent les trois règles tout en étant équivalentes.

Nous allons montrer, dans un premier temps, que toutes les classes d'équivalence ont un représentant dans \mathcal{S}_r .

Lemme 4.1. *Soit \mathcal{C} une configuration linéaire sur \mathcal{E} . Si \mathcal{C} n'appartient pas à \mathcal{S}_r alors il existe une configuration linéaire équivalente à \mathcal{C} qui appartient à \mathcal{S}_r .*

Preuve. Nous allons voir dans un premier temps comment passer de \mathcal{C} à une configuration de \mathcal{S}_r . On commence par appliquer à \mathcal{C} la permutation de \mathcal{E} qui transforme l'ordre \langle_1 de \mathcal{C} en $1 \langle_1 2 \langle_1 \dots \langle_1 r$. Ensuite pour chaque ordre \langle_o , on inverse \langle_o si $\text{ind}(1, \langle_i) < \lceil \frac{r}{2} \rceil$ ou si $\text{ind}(1, \langle_i) = \lceil \frac{r}{2} \rceil$ et $\text{ind}(2, \langle_i) < \text{ind}(1, \langle_i)$. Enfin on renumérote les ordres afin que les listes associées aux ordres soient triées selon l'ordre lexicographique. On note \mathcal{C}' la configuration obtenue.

Cette dernière étape nous assure que \mathcal{C}' respecte la Règle 4.2. Le fait de renuméroter les ordres n'a pas eu d'impact sur la position des éléments dans les différents ordres. Ainsi pour tout ordre 1 fait parti des $\lceil \frac{r}{2} \rceil$ plus petites valeurs et si r est impair et que 1 est la valeur médiane alors la valeur 2 est plus petite que 1 dans tous les ordres. \mathcal{C}' respecte donc la Règle 4.3. Afin d'obtenir \mathcal{C}' , nous avons commencé par transformer l'ordre \langle_1 de \mathcal{C} en $1 \langle_1 2 \langle_1 \dots \langle_1 r$.

L'élément 1 étant le plus petit dans l'ordre \langle_1 nous n'avons pas inversé l'ordre \langle_1 dans l'étape suivante. Ensuite nous avons renuméroté les ordres selon l'ordre lexicographique. Or le plus petit ordre possible selon l'ordre lexicographique étant l'ordre $1 \langle 2 \langle \dots \langle r$, après renumérotation des ordres on avait bien pour l'ordre \langle_1 , $1 \langle_1 2 \langle_1 \dots \langle_1 r$. Ainsi \mathcal{C}' respecte bien la Règle 4.1. Donc \mathcal{C}' appartient à \mathcal{S}_r .

Nous avons réalisé trois étapes pour passer de \mathcal{C} à \mathcal{C}' . La première consistait à fixer l'ordre \langle_1 comme étant égale à $1 \langle_1 2 \langle_1 \dots \langle_1 r$, ce qui revient à faire une permutation de \mathcal{E} . La deuxième étape consistait à inverser certains ordres. Enfin la troisième étape consistait à renuméroter les ordres, ce qui revient à faire une permutation de $\langle_1, \dots, \langle_{r-1}$. Ainsi les configurations \mathcal{C} et \mathcal{C}' sont égales, à une permutation de \mathcal{E} , une permutation de $\langle_1, \dots, \langle_{r-1}$ et des inversions d'ordres près. Ainsi \mathcal{C} possède une configuration équivalente dans \mathcal{S}_r . \square

Or, comme nous l'avons dit dans la remarque précédente, les deux configurations linéaires \mathcal{C}_1 et \mathcal{C}_2 de la Section 4.1 appartiennent à \mathcal{S}_r , la taille de \mathcal{S}_r n'est qu'une borne supérieur au nombre de configurations linéaires à équivalence près. Pour calculer cette borne supérieure, nous aurons besoin du lemme suivant :

Lemme 4.2. *Le nombre de multi-ensembles de k objets pris parmi s est égale au nombre de parties de k objets pris parmi $s + k - 1$.*

Preuve. Nous allons faire une bijection entre les multi-ensembles et les chemins dans une grille. On considère la grille $G_{k+1,s}$ composée de $k+1$ lignes et s colonnes. On numérote les lignes de 0 à k et les colonnes de 1 à s . Un multi-ensemble de k objets pris parmi s est un chemin dans $G_{k+1,s}$ du coin inférieur gauche (point $(0, 1)$) au coin supérieur droit (point (k, s)) en n'autorisant des déplacements uniquement de gauche à droite ou du bas vers le haut.

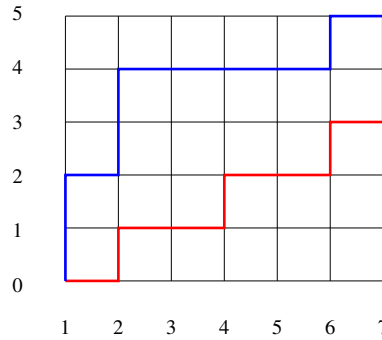


FIGURE 4.1 – Exemple d'une grille $G_{6,7}$. Le chemin en bleu correspond au multi-ensemble $\{1, 1, 2, 2, 6\}$ et le chemin en rouge correspond au multi-ensemble $\{2, 4, 6, 7, 7\}$.

En effet, chaque colonne correspond à un objet et chaque ligne correspond au nombre d'objets pris dans notre multi-ensemble. Dans notre chemin si l'on est au point (i, j) , alors le fait d'aller vers le haut (d'aller au point $(i+1, j)$) signifie que l'on rajoute un fois l'élément j à notre multi-ensemble. Par contre le fait d'aller vers la droite (d'aller au point $(i, j+1)$) correspond à ne plus rajouter l'élément j à notre multi-ensemble. La Figure 4.1 donne l'exemple d'une grille et de deux chemins correspondant à deux multi-ensembles.

Or un chemin dans $G_{k+1,s}$ pour aller du point $(0, 1)$ au point (k, s) en ne parcourant les segments de la grille que de gauche à droite ou du bas vers le haut correspond à un ensemble de k objets pris parmi $s+k-1$. En effet, un tel chemin utilise $s+k-1$ segments de la grille dont exactement k sont verticaux. Ainsi dans la Figure 4.1 le chemin bleu correspond à l'ensemble $\{1, 2, 4, 5, 10\}$ et le chemin rouge correspond à l'ensemble $\{2, 5, 8, 10, 11\}$.

Ainsi le nombre de multi-ensembles de k objets pris parmi s est égal au nombre d'ensembles de k objets pris parmi $s+k-1$. On a alors :

$$\binom{s}{k} = \binom{s+k-1}{k}$$

□

Nous allons maintenant donner une borne supérieure au nombre de configurations linéaires à équivalence près, qui correspond à la taille de l'ensemble de toutes les configurations respectant les trois règles.

Théorème 4.3. *Le nombre de configurations linéaires à équivalence près $C_{diff}(r)$ admet comme borne supérieure $\binom{\frac{r}{2}+(r-2)-1}{r-2}$.*

Preuve. D'après le Lemme 4.1, \mathcal{S}_r contient au moins une configuration linéaire pour chaque classe d'équivalence. Ainsi le cardinal de \mathcal{S}_r nous donne une borne supérieure pour $C_{diff}(r)$.

Nous allons calculer le cardinal de \mathcal{S}_r . Comme les ordres sur \mathcal{E} sont en bijection avec les permutations de $\{1, \dots, r\}$, il y a au total $r!$ ordres sur \mathcal{E} . Or la Règle 4.3 nous indique que nous n'allons utiliser que la moitié de ces ordres soit un total de $\frac{r!}{2}$ ordres.

Nous avons vu qu'une configuration est un multi-ensemble de $r-1$ ordres. Or la Règle 4.1 assure que l'ordre $<_1$ est égal à $1 <_1 2 <_1 \dots <_1 r$. Ainsi le cardinal de \mathcal{S}_r est égal au nombre de multi-ensembles de $r-2$ ordres parmi ces $\frac{r!}{2}$ ordres. D'après le Lemme 4.2 on a :

$$|\mathcal{S}_r| = \binom{\frac{r!}{2} + (r-2) - 1}{r-2}$$

□

4.2.2 Une borne inférieure

Nous allons voir que le cardinal de \mathcal{S}_r permet également d'obtenir une borne inférieure au nombre de configurations linéaires à équivalence près.

Théorème 4.4. Soient \mathcal{C} et \mathcal{C}' deux configurations différentes de \mathcal{S}_r sur \mathcal{E} . Soient $\sigma_1(<_i)$ et $\sigma_2(<_i)$ les permutations de $\{1, \dots, r\}$ qui transforme l'ordre $<_i$ respectivement en $1 <_i 2 <_i \dots <_i r$ et en $r <_i r-1 <_i \dots <_i 1$. \mathcal{C} est équivalente à \mathcal{C}' si et seulement si il existe un ordre $<_i$ de \mathcal{C} tel que l'on puisse passer de \mathcal{C} à \mathcal{C}' en appliquant la permutation $\sigma_1(<_i)$ ou la permutation $\sigma_2(<_i)$ à tous les ordres de \mathcal{C} puis en appliquant les Règles 4.2 et 4.3.

Preuve. \Leftarrow) Les Règles 4.2 et 4.3 correspondent respectivement à une permutation de $<_1, \dots, <_r$ et à des inversions d'ordres. Ainsi les configurations \mathcal{C} et \mathcal{C}' sont égales à une permutation de \mathcal{E} , une permutation de $<_1, \dots, <_r$ et à des inversions d'ordres près, elles sont donc équivalentes.

\Rightarrow) Si \mathcal{C} et \mathcal{C}' sont équivalentes il y a alors deux cas : soit une permutation de \mathcal{E} est nécessaire pour passer de \mathcal{C} à \mathcal{C}' , soit il n'est pas nécessaire de réaliser une permutation de \mathcal{E} pour passer de \mathcal{C} à \mathcal{C}' .

Supposons dans un premier temps qu'une permutation π de \mathcal{E} soit nécessaire. On note alors \mathcal{C}_π la configuration obtenue à partir de \mathcal{C} en appliquant la permutation π à tous les ordres de \mathcal{C} . Comme \mathcal{C}' respecte la Règle 4.1 l'ordre $<_1$ de \mathcal{C}' est égal à $1 <_1 2 <_1 \dots <_1 r$. Ainsi avant application des Règles 4.2 et 4.3, il existe un ordre $<_i$ de \mathcal{C}_π tel que l'on ait $1 <_i 2 <_i \dots <_i r$ ou $r <_i r-1 <_i \dots <_i 1$. Donc la permutation de \mathcal{E} est soit $\sigma_1(<_i)$ soit $\sigma_2(<_i)$.

Nous allons voir que si \mathcal{C} et \mathcal{C}' sont deux configurations équivalentes de \mathcal{S}_r sur \mathcal{E} telles que l'on puisse passer de \mathcal{C} à \mathcal{C}' sans réaliser de permutation de \mathcal{E} alors \mathcal{C} et \mathcal{C}' sont égales. \mathcal{C} respecte la Règle 4.3 donc si l'on inverse le moindre ordre de \mathcal{C} , la configuration obtenue ne respectera plus cette règle. Comme le fait de permuter les ordres de la configuration obtenue ne la fera pas respecter cette règle, cela prouve que pour passer de \mathcal{C} à \mathcal{C}' on n'inverse aucun ordre. Or \mathcal{C} respecte la Règle 4.2 donc si l'on fait une permutation d'ordres sur \mathcal{C} , la configuration obtenue ne respectera plus la Règle 4.2. Ainsi si \mathcal{C} et \mathcal{C}' sont deux configurations équivalentes de \mathcal{S}_r sur \mathcal{E} telles que l'on puisse passer de \mathcal{C} à \mathcal{C}' sans réaliser de permutation de \mathcal{E} , alors \mathcal{C} et \mathcal{C}' sont égales. □

Ce théorème va nous servir pour calculer la borne inférieure au nombre de configurations linéaires à équivalence près. Il nous servira aussi dans la Section 4.3.2 pour les algorithmes permettant de calculer le nombre de configurations linéaires à équivalence près.

Théorème 4.5. *Le nombre de configurations linéaires à équivalence près $C_{diff}(r)$ admet comme borne inférieure $\frac{1}{(2r-2)} \times \binom{\frac{r}{2}+(r-2)-1}{r-2}$.*

Preuve. D'après le Théorème 4.4, deux configurations linéaires \mathcal{C} et \mathcal{C}' différentes de \mathcal{S}_r sont équivalentes si et seulement si il existe un ordre $<_i$ de \mathcal{C} tel que l'on puisse passer de \mathcal{C} à \mathcal{C}' en appliquant la permutation $\sigma_1(<_i)$ ou la permutation $\sigma_2(<_i)$ à tous les ordres de \mathcal{C} puis en appliquant les Règles 4.2 et 4.3. Ainsi pour tout $i \in \{1, \dots, r-1\}$ chaque configuration linéaire de \mathcal{S}_r possède au plus 2 configurations linéaires équivalentes appartenant à \mathcal{S}_r (une en appliquant $\sigma_1(<_i)$ et une en appliquant $\sigma_2(<_i)$). Or appliquer la permutation $\sigma_1(<_1)$ ne change pas la configuration car l'ordre $<_1$ est déjà égal à $1 <_1 2 <_1 \dots <_1 r$ d'après la Règle 4.1. Ainsi chaque classe d'équivalence possède au plus $2(r-1)$ représentants dans \mathcal{S}_r . \square

4.3 Méthode de calcul de la valeur exacte

Comme nous n'avons pas réussi à trouver une formule exacte pour $C_{diff}(r)$, nous avons créé un programme pour calculer les valeurs de $C_{diff}(r)$ pour r petit. Le principe de ce programme est de générer les unes après les autres les configurations de \mathcal{S}_r . Pour chaque configuration générée, on teste de manière efficace s'il n'existe pas une configuration qui lui est équivalente parmi les configurations déjà générées. Les algorithmes décrits ci-dessous ont été implémentés en C++.

4.3.1 Recherche d'un ensemble d'ordres totaux équivalent

Nous verrons dans la suite comment créer un tableau des ordres respectant la Règle 4.3 et comment générer les configurations linéaires respectant les trois règles à partir de ce tableau. Pour l'instant nous allons voir comment rechercher, si une configuration possède une configuration équivalente parmi celles déjà générées.

Notre première idée était de stocker les configurations linéaires générées grâce à une liste ordonnée. Pour ce faire, à chaque fois qu'une configuration \mathcal{C} est générée nous associons à cette configuration linéaire une chaîne de caractères. Nous calculons alors toutes les configurations équivalentes à \mathcal{C} qui vérifient les trois règles (nous allons voir juste après comment les calculer) et leurs chaînes de caractères associées. Parmi toutes ces chaînes de caractères, nous gardons la chaîne de caractère qui est la plus petite (via l'ordre lexicographique). Ensuite nous cherchons cette chaîne de caractère dans la liste ordonnée. Cette chaîne de caractère est déjà présente dans la liste ordonnée si et seulement s'il existe une configuration linéaire déjà générée qui est équivalente à la configuration \mathcal{C} . Ainsi, nous ajoutons cette chaîne de caractères si elle n'est pas déjà présente en l'insérant au bon endroit de la liste de sorte que ces éléments soient toujours triés selon l'ordre lexicographique. Le problème de cette méthode est qu'elle est gourmande aussi bien en temps de calcul pour parcourir la liste, qu'en espace mémoire pour stocker la liste.

Pour améliorer la méthode nous avons voulu utiliser une table de hachage à la place d'une liste pour stocker les configurations. Le principe des tables de hachage est d'utiliser une fonction mathématique pour associer un entier à chaque élément que l'on souhaite stocker. Si la fonction est bien choisie, il n'y a que peu d'éléments associés à un même entier. On stocke alors chaque élément e à la case de la table ayant pour indice l'entier associé à e (une case peut contenir plusieurs éléments). Ainsi lorsque l'on cherche si un élément est dans la table de hachage, on calcule l'entier i associé à cet élément puis on le compare à tous les éléments présents à la case i ce qui diminue le temps de calcul dû au parcours de la liste. Le problème des tables de hachages est que leur utilisation ne réduit pas l'espace mémoire nécessaire (il est même très légèrement augmenté).

Position d'une configuration linéaire

A cause de la limite d'espace mémoire, nous n'avons réussi qu'à obtenir le nombre de configurations linéaires à équivalence près pour $r \leq 5$ avec les méthodes précédentes. Pour obtenir la valeur suivante il fallait changer complètement la méthode.

Le principe de la nouvelle méthode repose sur le fait que les configurations linéaires sont générées dans un certain ordre. Nous avons alors eu l'idée de calculer, pour chaque configuration linéaire \mathcal{C} , la position de \mathcal{C} dans l'ordre de génération des configurations linéaires (on donnera la définition formelle plus tard). Ainsi lorsque l'on génère une configuration linéaire \mathcal{C}_G , on calcule la position de \mathcal{C}_G (notée $p_c(\mathcal{C}_G)$). Ensuite on crée les configurations équivalentes à \mathcal{C}_G et on calcule leurs positions. Si une de ces positions est strictement inférieure à $p_c(\mathcal{C}_G)$ alors la configuration linéaire \mathcal{C}_G possède une configuration linéaire équivalente parmi celles déjà générées.

Calcul de la position d'une permutation. Afin d'obtenir la position d'une configuration nous avons besoin de calculer la position d'une permutation de $\{1, \dots, k\}$ dans l'ordre lexicographique.

Soit π_k une permutation de $\{1, \dots, k\}$. On définit la position de la permutation π_k (noté $p_p(\pi_k)$) comme étant le nombre de permutations de $\{1, \dots, k\}$ strictement inférieures à π_k dans l'ordre lexicographique. Comme le nombre de permutations de $\{1, \dots, k\}$ est égal à $k!$, la position de la permutation π_k (noté $p_p(\pi_k)$) est un entier appartenant à $\{0, \dots, k! - 1\}$.

On code π_k par un tableau T de taille k et on note $T[i]$ l'élément du tableau à la case d'indice i . Nous allons calculer la position de π_k par récurrence sur la taille du tableau T . Si le tableau n'a qu'une seule case, on doit compter le nombre de permutations de taille 1 qui sont strictement inférieure à T , ce qui correspond au nombre d'entiers de $\{1, \dots, k\}$ qui sont strictement inférieurs à $T[0]$. On retourne donc la valeur $T[0] - 1$.

Si le tableau T a au moins deux cases, on va traiter d'un côté la première case puis travailler sur le reste du tableau. Grâce à la première case de T , on peut compter le nombre de permutations dont le premier élément est strictement inférieur à $T[0]$. Pour chaque entier strictement inférieur à $T[0]$ il y a exactement $(k-1)!$ permutations de $\{1, \dots, k\}$ commençant par cet entier. Ainsi on ajoutera au résultat trouvé pour le reste du tableau, le nombre de permutations de $\{1, \dots, k-1\}$ commençant par un élément strictement inférieur à $T[0]$, c'est-à-dire $(k-1)! \times (T[0] - 1)$. Pour calculer le résultat sur le reste du tableau, il faut que les $k-1$ dernières cases forment une permutation. On soustrait 1 à toutes les valeurs qui sont supérieur à $T[0]$ puis on supprime la première case. On obtient alors un tableau de $k-1$ cases qui est une permutation de $\{1, \dots, k-1\}$ sur lequel on utilise la même méthode.

position
d'une per-
mutation

D'après ce que nous venons de voir, lorsqu'on travaille sur la case i de T , on s'intéresse au nombre total de permutations de $\{1, \dots, k-i\}$. Il y en a donc $(k-i)!$. De plus à chaque étape j de la récurrence (i.e. lorsqu'on travaille sur la case $j-1$ de T), on diminue de 1 la valeur de $T[i]$ si $i > j-1$ et que $T[i] > T[j-1]$. Ainsi pour chaque case i on diminue de 1 la valeur de $T[i]$ pour toute cases de T dont l'indice est strictement inférieur à i et la valeur est strictement inférieur à $T[i]$. En notant $\alpha(i)$ le nombre de cases de T dont l'indice est strictement inférieur à i et la valeur est strictement inférieur à $T[i]$, le fait d'appliquer la méthode que nous venons de décrire revient à appliquer la formule suivante :

$$p_p(\pi_k) = \sum_{i=0}^{k-1} (k-i-1)! \times (T[i] - \alpha(i) - 1)$$

Calcul de la position d'un ordre. Pour pouvoir calculer la position d'une configuration linéaire, il faut dans un premier temps calculer la position d'un ordre linéaire respectant la Règle 4.3. Soient E_r l'ensemble de tous les ordres linéaires sur $\{1, \dots, r\}$ respectant la Règle 4.3 et $<_o$ un ordre de E_r . On définit la *position de l'ordre* $<_o$ (noté $p_o(<_o)$) comme étant le nombre d'ordres de E_r strictement inférieurs à $<_o$ selon l'ordre lexicographique.

position
d'un ordre

Il est important de noter que bien qu'un ordre corresponde à une permutation, la position d'un ordre est différente de la position de la permutation correspondante. En effet, il y a $r!/2$ ordres linéaire respectant la Règle 4.3 alors qu'il y a $r!$ permutations. Ainsi, si le premier élément d'un ordre est r alors la position de cet ordre sera strictement inférieure à $r!/2$ alors que la position de la permutation correspondant à cet ordre sera au moins de $(r-1)! \times (r-1)$ ($(r-1)! \times (r-1) > r!/2$ pour $r > 2$).

Le cas où l'élément 1 est l'élément médian d'un ordre est plus délicat. En effet, si dans un ordre $<_o$ on supprime l'élément 1, on obtient une partition de $\{2, \dots, r\}$. Si dans $<_o$, l'élément 1 n'est pas l'élément médian alors on peut obtenir en supprimant 1 n'importe quelle partition de $\{2, \dots, r\}$. A l'inverse, si 1 est l'élément médian de $<_o$ alors en supprimant 1 on obtient une permutation de $\{2, \dots, r\}$ dans laquelle l'indice de l'élément 2 est inférieur à $(r-2)/2$ (comme il s'agit d'une permutation entre $r-1$ entiers, l'indice d'un élément est compris entre 0 et $r-2$).

Nous allons supposer dans un premier temps que r est pair ou que $ind(1, <_o) < (r-1)/2$. Le calcul de la position de $<_o$ se passe en deux temps. On s'intéresse tout d'abord à l'indice de l'élément 1 dans $<_o$. Le nombre total d'ordres ayant l'élément 1 dans une position bien précise est égal au nombre de permutations de $r-1$ éléments, soit $(r-1)!$. Il faut alors compter le nombre d'ordres dont l'élément 1 est placé avant la case d'indice $ind(1, <_o)$. En ne s'intéressant qu'à l'indice de 1, la position de $<_o$ vaut alors $(r-1)! \times ind(1, <_o)$. A cette valeur il faut maintenant ajouter la position de l'ordre $<_o$ parmi tous les ordres ayant l'élément 1 à la case d'indice $ind(1, <_o)$. Pour cela on crée un tableau T_o à une dimension de $r-1$ cases dans lequel on stocke dans l'ordre tous les éléments de $<_o$ sauf l'élément 1. Ensuite on soustrait 1 à tous les éléments du tableau T_o pour que ce tableau corresponde à une permutation de $\{1, \dots, r-1\}$. On ajoute alors à la valeur de $p_o(<_o)$ la valeur de la position de la permutation stocker dans T_o .

Si 1 est l'élément médian de $<_o$ on a alors que l'indice de 1 est égal à $(r-1)/2$ et l'indice de 2 est strictement inférieur à $(r-1)/2$. On commence alors par compter le nombre d'ordres dont l'élément 1 est placé avant la case d'indice $(r-1)/2$, il y en a $(r-1)! \times (r-1)/2$. Puis on ajoute le

nombre d'ordres dont l'élément 1 est à la case d'indice $(r-1)/2$ et l'élément 2 est placé avant la case d'indice $ind(2, <_o)$, c'est-à-dire $(r-2)! \times ind(2, <_o)$. On crée ensuite un tableau T_o à une dimension de $r-2$ cases dans lequel on stocke dans l'ordre tous les éléments de $<_o$ sauf les éléments 1 et 2. On soustrait 2 à tous les éléments du tableau T_o pour que ce tableau corresponde à une permutation de $\{1, \dots, r-2\}$. On ajoute alors à la valeur de $p_o(<_o)$ la valeur de la position de la permutation stocker dans T_o .

On a alors :

$$p_o(<_o) = \begin{cases} (r-1)! \cdot \frac{(r-1)}{2} + (r-2)! \cdot ind(2, <_o) + p_p(T_o) & \text{si 1 est l'élément médian de } <_o \\ (r-1)! \cdot ind(1, <_o) + p_p(T_o) & \text{sinon} \end{cases}$$

En stockant dans un tableau les valeurs des factorielles, ce calcul se fait alors en $O(r^2)$ car à chaque étape on parcourt le tableau pour soustraire 1 (ou 2 si 1 est l'élément médian) à certaines cases du tableau.

Calcul de la position d'une configuration linéaire. Le calcul de la position d'une configuration linéaire se base sur le fait qu'une configuration linéaire est un multi-ensemble de $r-1$ ordres parmi les $r!/2$ ordres respectant la Règle 4.3. Pour calculer la position d'un multi-ensemble de k éléments parmi s on procède de la façon suivante : on représente un multi-ensemble par un tableau S dont les éléments sont rangés par ordre croissant ($\forall i, j \in \{0, \dots, k-1\}, i < j$ on a $S[i] \leq S[j]$). On note alors $p_m(S)$ la position de ce multi-ensemble. Pour le calcul de $p_m(S)$, on commence par s'occuper de la première case du tableau. La position du multi-ensemble est alors égale au nombre de multi-ensembles à $k-1$ éléments dont le premier élément est strictement inférieur à $S[0]$, soit $\sum_{i=0}^{S[0]-1} \binom{s-i}{k-1}$. On procède de la même manière pour les autres cases du tableau. Ainsi pour une case $j \neq 0$, on compte le nombre de multi-ensembles à $k-j-1$ éléments dont les éléments des cases 0 à $j-1$ sont les mêmes que pour S et dont l'élément à la case j appartient à $\{S[j-1], \dots, S[j]-1\}$. Pour une case $j \neq 0$, il y en a $\sum_{i=S[j-1]}^{S[j]-1} \binom{s-i}{k-j-1}$. Ainsi la formule pour calculer la position $p_m(S)$ est :

$$p_m(S) = \sum_{j=0}^{k-1} \sum_{i=min}^{S[j]-1} \binom{s-i}{k-j-1}$$

où $min = 0$ si $j = 0$ et $min = S[j-1]$ sinon.

En utilisant la formule du Lemme 4.2 on obtient :

$$p_m(S) = \sum_{j=0}^{k-1} \sum_{i=min}^{S[j]-1} \binom{(s-i) + (k-j-1) - 1}{k-j-1}$$

Ainsi pour calculer la position d'une configuration linéaire \mathcal{C} , on utilise cette formule où S est constitué des positions des ordres de \mathcal{C} , $k = r-1$ et $s = r!/2$. Dans cette formule, on calcule exactement $S[k-1] - 1$ coefficients binomiaux (on calcule des coefficients binomiaux pour i allant de 0 à $S[0] - 1$ puis pour i allant de $S[0]$ à $S[1] - 1$ et ainsi de suite jusqu'à i allant de $S[k-2]$ à $S[k-1] - 1$). Or pour calculer un coefficient binomial on se base sur le principe suivant : comme

$\binom{n}{p} = \binom{n}{n-p}$, on ne calcule les coefficients que pour $p \leq n/2$ et pour cela on utilise la formule $\binom{n}{p} = (n-k+1) \prod_{i=2}^p (n-p+i)/i$. Ainsi chaque calcul de coefficient binomial se fait en $O(p)$ opérations (avec pour nous $p = k-j-1 = (r-1) - j - 1$). Or pour tout multi-ensemble $S[k-1] < s = r!/2$. Ainsi la position d'une configuration linéaire se fait en $O(r! \times r)$ opérations.

En pratique cette méthode est bien plus efficace car même si pour chaque configuration linéaire on réalise $O(r! \times r)$ opérations, on évite la recherche parmi les $\binom{\frac{n}{2} + (r-2) - 1}{r-2}$ configurations linéaires générées grâce aux Règles 4.1, 4.2 et 4.3. On évite ainsi de créer une structure de taille $O(r^{r'})$ pour stocker les configurations linéaire générées.

4.3.2 Algorithmes pour calculer la valeur exacte

Soit \mathcal{C} une configuration respectant les trois règles énoncées précédemment.

D'après le Théorème 4.4, pour calculer toutes les configurations équivalentes à \mathcal{C} qui respectent les trois Règles, il suffit de calculer les configurations créées par les permutations transformant chaque ordre de \mathcal{C} en $1 <_i 2 <_i \dots <_i r$ puis en $r <_i r-1 <_i \dots <_i 1$ et d'appliquer à ces configurations les Règles 4.2 et 4.3. Ainsi pour chaque configuration linéaire respectant les trois règles, il existe au plus $2r-1$ autres configurations linéaires équivalentes respectant ces règles. L'Algorithme 4.1 applique ce principe et utilise le calcul la position d'une configuration linéaire pour rechercher si une configuration donnée possède une équivalente déjà générée.

Algorithme 4.1 : Recherche d'une configuration équivalente déjà générée

Entrée : Une configuration linéaire \mathcal{C} sous forme d'un tableau respectant les trois règles.

Sortie : Vrai si la configuration \mathcal{C} est équivalente à une configuration linéaire déjà générée, Faux sinon.

Créer un tableau O tel que O_1 soit égal à $1, 2, \dots, r$ et O_2 soit égal à $r, r-1, \dots, 1$;

Créer un tableau \mathcal{C}_{eq} ;

Calculer la position de \mathcal{C} et la stocker dans un entier min ;

$trouve \leftarrow$ FAUX;

$i \leftarrow 1$;

tant que $non(trouve)$ **ET** $i \leq r-1$ **faire**

$j \leftarrow 1$;

tant que $non(trouve)$ **ET** $j \leq 2$ **faire**

 Copier \mathcal{C} dans \mathcal{C}_{eq} ;

 Calculer la permutation qui transforme $\mathcal{C}[i]$ en O_j ;

 Appliquer cette permutation à \mathcal{C}_{eq} ;

 Inverser les ordres de \mathcal{C}_{eq} qui ne respectent pas la Règle 4.3;

 Permuter les ordres de \mathcal{C}_{eq} pour qu'ils respectent la Règle 4.2;

 Calculer la position de \mathcal{C}_{eq} et la stocker dans pos_{eq} ;

si $pos_{eq} < min$ **alors**

$trouve \leftarrow$ VRAI;

$j \leftarrow j + 1$;

$i \leftarrow i + 1$;

Génération de l'ensemble des configurations de \mathcal{S}_r

Maintenant que nous savons comment déterminer si une configuration possède une configuration équivalente déjà générée, il ne nous reste plus qu'à voir comment générer les configurations de \mathcal{S}_r . Afin de générer l'ensemble de ces configurations (i.e. l'ensemble des configurations linéaires respectant les Règles 4.1, 4.2 et 4.3), nous allons commencer par créer un tableau contenant les ordres utilisés dans ces configurations. Nous pourrons alors utiliser ce tableau pour générer les configurations linéaires respectant ces règles.

Algorithme 4.2 : Création du tableau des ordres à utiliser

Entrée : Un entier r correspondant au nombre d'éléments des ordres d'une configuration.

Sortie : Le tableau des ordres à utiliser dans la création des configurations.

Créer le tableau *ordres* de taille $\frac{r!}{2} \times r$;

Créer un tableau *permut* égal à $\{2, 3, \dots, r\}$;

$k \leftarrow \lfloor \frac{r}{2} \rfloor$;

$j \leftarrow -1$;

for $pos_1 = 1$ à k **do**

répéter

$j \leftarrow j + 1$;

for $i = 1$ to $pos_1 - 1$ **do**

$ordre_{j,i} \leftarrow permut_i$;

$ordre_{j,pos_1} \leftarrow 1$;

for $i = pos_1 + 1$ to r **do**

$ordre_{j,i} \leftarrow permut_{i-1}$;

jusqu'à $next_permutation(permut_2, permut_2 + (r - 2)) = FAUX$;

Création du tableau des ordres à utiliser. L'Algorithme 4.2 indique comment créer ce tableau. Dans cet algorithme, on crée dans un premier temps un tableau à deux dimensions de taille $\frac{r!}{2} \times r$ où chaque ligne du tableau correspond à un ordre respectant la Règle 4.3. Notons pos_1 la position de 1 et pos_2 la position de 2 dans l'ordre que nous allons créer. Il faut donc traiter différemment le cas où r est pair et le cas où r impair. Le cas le plus simple est le cas où r est pair. On doit alors faire varier pos_1 de 1 à $r/2$. Le reste de l'ordre correspond à une permutation de $(2, \dots, r)$. En C++, pour créer facilement toutes les permutations de $(2, \dots, r)$, il suffit d'utiliser la fonction $next_permutation(tab, tab + (r - 1))$ qui prend les $r - 1$ premiers éléments d'un tableau unidimensionnel tab , qui remplit le tableau tab avec la permutation suivante dans l'ordre lexicographique (si elle existe) et qui renvoie vrai (ou faux si elle n'existe pas). Si r est impair alors il faut créer en plus les ordres où $pos_1 = (r + 1)/2$ et $pos_2 < (r + 1)/2$. Pour les créer il suffit de faire varier pos_2 de 1 à $(r - 1)/2$ et le reste de l'ordre correspond à une permutation de $(3, \dots, r)$.

Génération des configurations linéaires respectant les trois règles. Cette partie du programme est la plus simple. Comme nous l'avons vu lors du calcul de la borne supérieure, il suffit en effet de générer toutes les combinaisons avec répétition de $r - 2$ ordres parmi les $\frac{r!}{2}$ ordres du tableau précédent. L'Algorithme 4.3 permet de générer toutes ces combinaisons. On commence par créer

un tableau $comb$ de taille $r - 2$ dont toutes les valeurs sont initialisées à 0 et on initialise un entier i à 1 qui servira à se déplacer dans le tableau $comb$. A chaque étape on augmente de 1 la valeur de $comb_i$ puis la valeur de i . Ensuite si on a $i = (r - 1)$ alors on a rempli tout le tableau et ce tableau correspond à une combinaison avec répétition. On peut alors se servir de $comb$ pour créer une configuration linéaire. On diminue i de 1 pour continuer à travailler sur la dernière case du tableau. Puis si la valeur de $comb_i$ vaut $\frac{r!}{2}$, cela signifie que l'on a fini de travailler avec cette case et qu'il faut revenir à une case précédente du tableau. On cherche alors la dernière case du tableau ne valant pas $\frac{(r+1)!}{2}$. S'il n'y en a pas, on a alors terminé de lister toutes les combinaisons avec répétition. Sinon on donne à i l'indice de cette case, et pour toutes les cases dont l'indice est plus grand que i on donne comme valeur la valeur de $comb_i$. Ainsi aux étapes suivantes on va augmenter de 1 la valeur de toutes les cases d'indice i à $r - 2$ ce qui nous donnera la combinaison avec répétition suivante.

Algorithme 4.3 : Génération de toutes les configurations respectant les trois règles

Entrée : Le nombre d'ordres r d'une configuration linéaire et le tableau $ordres$ créé par l'Algorithme 4.2.

Sortie : L'ensemble S des configurations linéaires à équivalence près (en utilisant l'Algorithme 4.1).

Créer le tableau $comb$ de taille $r - 2$ dont toutes les valeurs sont initialisées à 0;

$S \leftarrow \emptyset$;

$i \leftarrow 0$;

$arret \leftarrow \text{FAUX}$;

tant que $arret = \text{FAUX}$ **faire**

$comb_i \leftarrow comb_i + 1$;

$i \leftarrow i + 1$;

si $i = n$ **alors**

$k := comb_i$;

 Créer la configuration \mathcal{C} dont le i -ème ordre correspond à $ordre_k$;

 Utiliser l'Algorithme 4.1 pour ajouter \mathcal{C} à S s'il n'existe pas dans S une configuration équivalente à \mathcal{C} ;

$i \leftarrow i - 1$;

si $comb_i = \frac{r!}{2}$ **alors**

tant que $comb_i = \frac{r!}{2}$ **ET** $i \geq 1$ **faire**

$i \leftarrow i - 1$;

si $i = 0$ **alors**

$arret \leftarrow \text{VRAI}$;

sinon

pour $j = i$ **to** $r - 1$ **faire**

$comb_j \leftarrow comb_j$;

4.3.3 Comparaison entre la valeur exacte et son encadrement

Le programme précédent a été utilisé pour calculer la valeur de $C_{diff}(r)$ pour $r \leq 6$. Le Tableau 4.1 indique la valeur de $C_{diff}(r)$ ainsi que celle de sa borne supérieure $|\mathcal{S}_r|$ et de celle de sa borne inférieure $|\mathcal{S}_r|/(2r-3)$ en fonction de r .

r	2	3	4	5	6	7
$C_{diff}(r)$	1	2	21	5 097	71 965 235	?
$ \mathcal{S}_r $	1	2	78	37 820	711 563 490	$8,5024449 \times 10^{14}$
$ \mathcal{S}_r /(2r-2)$	1	1	13	4 728	71 156 349	$7,0853707 \times 10^{13}$

TABLE 4.1 – valeur de $C_{diff}(r)$ et de sa borne supérieure

Cette suite d'entiers n'existait pas dans "The On-Line Encyclopedia of Integer Sequences". Nous l'avons donc ajoutée [22].

Les calculs n'ont pas été possibles pour des valeurs plus grandes à cause de la taille de \mathcal{S}_r c'est-à-dire du nombre trop important de configurations à générer (pour rappel ces configurations sont celles respectant les Règles 4.1, 4.2 et 4.3). En effet, le calcul du nombre de configurations linéaires à équivalence près pour $r = 6$ a déjà pris près de trois jours sur un ordinateur standard.

Par contre ce calcul peut être facilement parallélisé. Pour chaque configuration linéaire de \mathcal{S}_r il faut calculer l'indice de cette configuration et l'indice de ses configurations équivalentes. Or le calcul de l'indice d'une configuration linéaire ne dépend que de cette configuration. Le seul point qui pourrait poser problème pour paralléliser serait la génération des configuration linéaire de \mathcal{S}_r . Le plus simple pour générer ces configurations est de les générer successivement, mais il est facile de créer un processus qui à partir de la première configuration génère les k premières configurations et d'avoir un deuxième processus qui calcule les k configurations suivantes à partir de la $k + 1$ -ème configuration et ainsi de suite.



Partie III. Classification et caractérisation de chirotopes

Dans cette partie nous nous intéressons aux problèmes de la classification de chirotopes et de la caractérisation de classes de chirotopes (voir Chapitre 1, Section 1.2. Le problème de classification consiste à trouver une partition des chirotopes en classes alors que le problème de caractérisation est de trouver pour toute classe, des propriétés partagées par tous ses éléments et uniquement par eux (autrement dit, les éléments des autres classes ne possèdent pas ces propriétés).

Bien entendu, les problèmes de classification et de caractérisation ne concernent pas que la morphométrie 3D. En effet, avec le développement de l'informatique, le nombre et la taille des bases de données ont explosé et la classification et la caractérisation de données interviennent dans de nombreuses applications. Par exemple, au sein des entreprises commerciales, les données concernant les clients sont analysées et les clients sont classés en fonction des produits qu'ils achètent. Les entreprises peuvent ainsi proposer leurs produits seulement aux clients susceptibles de les acheter. Dans le domaine de la médecine, les résultats de la classification et de la caractérisation de données médicales permettent d'apporter une aide au diagnostic ou de repérer dans la population des personnes présentant des risques de déclencher certaines maladies.

Lorsque l'on souhaite classer (respectivement caractériser) des données, il y a deux éléments à prendre en compte : le type de données et l'objectif de la classification (resp. caractérisation).

Les données peuvent être de trois types : numériques, binaires ou nominales (aussi appelées qualitatives).

Dans les *données numériques* peuvent être soit continues soit discrètes. Les données sont dites continues si elles forment un sous-ensemble de \mathbb{R} comme par exemple la taille, le poids ou les coordonnées de points. A l'inverse, les données discrètes forment un sous-ensemble de \mathbb{N} (voire de \mathbb{Z}) comme par exemple un nombre de personnes ou d'objets, ou l'année de naissance. Les données numériques sont des données très pratiques puisqu'elles sont ordonnées (on peut les comparer par la relation d'ordre naturel sur \mathbb{R}) et elles permettent l'utilisation des outils statistiques habituels.

Les *données binaires* servent notamment à représenter les résultats de tests. On code le plus souvent les variables binaires par *true* et *false* ou par 0 et 1. On qualifie une variable binaire d'asymétrique si pour l'ensemble des échantillons il y a une valeur qui est présente en majorité et de

données
numé-
riques

données
binaires

symétrique si les deux valeurs sont présentes en quantité similaire. Par exemple si l'on étudie la population française, il y a environ 52% de femme et 48% d'hommes ainsi la variable représentant le sexe est symétrique. Par contre, une variable représentant si une personne est majeure ou mineure est asymétrique car seulement 20 à 25% de la population française est mineure. Ainsi deux personnes mineures seront plus "proches" que deux personnes majeures.

données
nominales

Dans les *données nominales*, les variables sont représentés par des noms ou des nombres ne pouvant pas être comparés directement par une distance classique ; comme par exemple les couleurs d'un objet ou le numéro de département de naissance.

Les objectifs de la classification et de la caractérisation peuvent être très variés et dépendent du domaine d'application. Il peut s'agir de rechercher des séquences communes (comme dans l'analyse du génome par exemple), d'obtenir des classes denses ou connectées, de réduire la taille d'un jeu de données en perdant un minimum d'information (par exemple par projection sur un espace de dimension inférieure), de séparer le plus possible les classes ou à l'inverse de repérer si des classes sont proches.

Pour répondre à tous ces besoins, de nombreuses méthodes statistiques ont été créées dans le domaine du Data Mining (ou fouille de données). Certaines sont spécifiques à un type de données ou cherche à atteindre un objectif particulier. Ainsi, il est couramment admis qu'il n'existe pas une méthode meilleure que les autres [86].

classification
supervisée

Les méthodes de classifications sont réparties en deux types : la *classification supervisée* et la *clustering* (ou *classification non-supervisée*). Ces deux types de classifications diffèrent sur les connaissances que nous avons sur les données.

Une classification supervisée est utilisée lorsqu'une partie des données est déjà classée. Plus précisément, il faut alors que toutes les classes soient définies et que l'on ait au moins un échantillon par classe. Le but de la classification supervisée est donc de classer l'autre partie des données. Cette méthode se déroule en deux temps. Dans un premier temps, l'algorithme passe par une phase d'apprentissage c'est-à-dire qu'il cherche ce qui caractérise les classes au sein des données. Dans un deuxième temps, l'algorithme utilise cette caractérisation pour classer les éléments qui n'ont pas été utilisés pendant la phase d'apprentissage. Si au départ tous les éléments sont classés, ces algorithmes permettent donc d'obtenir une caractérisation. Par exemple, imaginons qu'un biologiste ait mis au point une base de données concernant des chats. On peut imaginer que le zoologiste connaisse à quelle race appartiennent certains chats. S'il reçoit des données sur un autre chat dont il ne connaît pas la race, il utilisera une méthode de classification supervisée pour déterminer la race uniquement à l'aide des données.

clustering

Au contraire, un algorithme de *clustering* n'utilise pas de phase d'apprentissage et n'a donc pas besoin que des échantillons aient déjà été classés. A partir des données, l'algorithme regroupe les échantillons en groupes appelés clusters, dans le but de minimiser (ou maximiser) un certain critère. Ce critère peut par exemple utiliser des notions de distances, des notions de densité et de connectivité ou des lois de distribution.

Dans la classification non-supervisée, on utilise le terme de cluster au lieu de classe car il est impossible de savoir à quoi correspondent réellement ces clusters sans avoir analysé les résultats. Reprenons l'exemple sur les chats. Si notre base de données est composée de quatre races (persan, siamois, abyssin et chartreux) et que certains échantillons sont déjà classés alors un algorithme de classification supervisée va répartir les chats parmi ces races. Par contre il se peut qu'un algorithme

de clustering regroupe les échantillons en un nombre de clusters différent du nombre de races ou alors en quatre clusters qui n'auront peut être rien à voir avec les races.

Notre but est donc de classifier un ensemble de chirotopes puis de caractériser chaque classe (i.e. de trouver des propriétés partagées par tous les chirotopes de la classe) et de les discriminer (i.e. de trouver des propriétés partagées par tous les chirotopes de la classe de sorte que tout chirotope d'une autre classe ne vérifie pas ces propriétés). Dans le Chapitre 5, nous étudions certaines méthodes de classification déjà existantes afin de déterminer si elles peuvent être appliquées à un ensemble de matroïdes orientés et si elles peuvent nous fournir une caractérisation des classes de matroïdes orientés. Nous proposons ensuite dans le Chapitre 6 notre propre méthode de classification supervisée. Cette méthode utilise la notion de $(d+1)$ -uplets discriminants entre deux ensembles de chirotopes. Dans ce chapitre, nous présentons notre méthode dans le cas uniforme puis nous en donnons une généralisation dans le cas non-uniforme.

Analyse de méthodes de classification existantes

Dans ce chapitre nous nous intéressons aux méthodes de classification existantes. Comme il existe un grand nombre de méthodes de classification, nous n'étudierons que quelques méthodes parmi les plus classiques. Pour pouvoir utiliser ces méthodes il nous faudra convertir dans un premier temps les chirotopes de modèles en deux ou trois dimensions en données de type binaire ou numérique. Nous présenterons ensuite certaines méthodes classiques de clustering puis de classification supervisée. Nous expliquerons les avantages et les inconvénients de chaque méthode lorsqu'elles sont utilisées pour la classification de chirotopes issues de formes en trois dimensions et nous verrons si elles permettent d'obtenir une caractérisation des classes.

5.1 Transformer un chirotope en données de différents types

De nombreuses méthodes de classification sont spécifiques à un type de données : certaines ne fonctionnent qu'avec des données binaires alors que d'autres nécessitent des données numériques. Il nous faut donc préalablement transformer nos chirotopes en données nominales, binaires ou numériques. Lors de cette transformation il faut s'assurer de ne pas perdre d'informations mais également que les distances que nous allons utiliser sur les différents types de données soient équivalentes (au sens mathématique du terme). En effet, si par exemple on cherche dans un jeu de données quels sont les échantillons les plus proches avec deux distances qui ne sont pas équivalentes, alors les résultats peuvent être très différents. Ainsi si l'on veut pouvoir comparer les résultats de deux méthodes, il faut s'assurer que les distances utilisées dans les méthodes sont équivalentes.

Dans la suite de ce chapitre nous allons comparer et classer un ensemble de chirotopes. Nous noterons n le nombre de chirotopes à classer. D'après la définition des chirotopes (c.f. Section 2.1), les chirotopes sont des vecteurs d'éléments de $\{+, -, 0\}$. Ainsi, ce sont des données de type nominal et nous allons voir comment les transformer en données binaires ou numériques.

5.1.1 Transformation en données binaires

Nous allons transformer un chirotope d'un modèle en deux ou trois dimensions en un vecteur binaire. Par souci de simplicité, nous noterons par 0 et 1 les deux valeurs que peuvent prendre les variables binaires.

Dans le cas uniforme les chirotopes sont des vecteurs d'éléments de $\{+, -\}$. Ainsi ils correspondent déjà à des données binaires. On remplacera alors la valeur $+$ par 1 et $-$ par 0.

Par contre dans le cas non-uniforme nous sommes bien obligés de réaliser une transformation. La méthode classique pour transformer des données nominales en données binaires est la suivante. Pour chaque variable nominale ν on crée un nombre de variables binaires égal au nombre de valeurs que peut prendre ν . Ainsi la i -ème variable binaire indique si la variable ν prend ou non la i -ème valeur. On a alors que pour tout échantillon, parmi toutes les variables binaires correspondant à la variable ν une seule variable a pour valeur 1 et cette variable binaire correspond à la valeur de la variable ν dans cet échantillon.

Dans le cas non-uniforme, lorsque l'on applique cette méthode au chirotope χ , on crée pour tout $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ trois variables binaires B_- , B_0 et B_+ . Parmi les trois variables B_- , B_0 et B_+ , la seule valant 1 correspond au signe $\chi(B)$. On obtient alors un vecteur trois fois plus grand dans lequel deux variables sur trois vaut 0.

Comparaisons de données binaires. Pour calculer les distances entre des échantillons de type binaires, on peut bien sûr utiliser la distance euclidienne ou toute autre distance utilisée pour les données numériques (voir section suivante) mais la distance la plus utilisée si les variables sont symétriques est la *distance de Hamming*. Cette distance consiste à compter le nombre de variables différentes entre les deux échantillons. Formellement en notant d_{Ham} la distance de Hamming et l la taille des vecteurs binaires, on a :

$$d_{Ham}(x_a, x_b) = \#\{j \in \{1, \dots, l\} | x_a(j) \neq x_b(j)\}.$$

La distance de Hamming correspond à la distance L_1 si les variables sont codées par exemple par 0 et 1.

Dans le cas uniforme la distance de Hamming est exactement la distance entre deux chirotopes que nous avons définie à la Section 2.3. Dans le cas non-uniforme, pour la même raison que dans le paragraphe précédent, un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ ayant un signe différent dans les deux chirotopes est comptée deux fois dans le calcul de la distance de Hamming. Ainsi dans le cas non-uniforme, la distance de Hamming est égale à deux fois la distance définie à la Section 2.3.

Dans le cas de variables asymétriques, on souhaite généralement dire que deux échantillons sont similaires pour une variable donnée si la variable vaut 1 dans les deux échantillons. Par exemple si les données représentent la présence (par la valeur 1) ou l'absence de symptômes pour des patients d'un hôpital, alors on souhaite que deux patients présentant un même symptôme soient considérés comme étant similaires mais que deux patients qui n'ont ni l'un ni l'autre un symptôme donné ne soient pas forcément considérés comme similaires. De nombreux indices de similarité ont été définis en biologie (et plus précisément en taxinomie). De tels indices s'appuient

sur un tableau de contingence qui se présente comme ceci :

		x_b	
		1	0
x_a	1	$n_{11}(x_a, x_b)$	$n_{10}(x_a, x_b)$
	0	$n_{01}(x_a, x_b)$	$n_{00}(x_a, x_b)$

où $n_{ij}(x_a, x_b)$ est le nombre de variables valant i pour x_a et j pour x_b . Pour simplifier les notations, s'il n'y a pas d'ambiguïté, on notera n_{ij} au lieu de $n_{ij}(x_a, x_b)$. En notant l la taille des vecteurs, on a alors $l = n_{00} + n_{01} + n_{10} + n_{11}$.

Grâce à ce tableau on peut définir plusieurs indices de similarité tels que :

- l'indice de similarité de Jaccard [33] : $S_j(x_a, x_b) = n_{11} / (n_{11} + n_{10} + n_{01})$
- l'indice de similarité de Sokal et Sneath [73] : $S_{ss}(x_a, x_b) = (n_{11} + n_{00}) / (n_{10} + n_{01})$
- l'indice de similarité de Russel et Rao [66] : $S_{rr}(x_a, x_b) = n_{11} / (n_{11} + n_{10} + n_{01} + n_{00})$

De nombreux autres indices de similarité sont listés notamment par Cheetham et Hazel [11] et par Nakache et Confais [49]. Cheetham et Hazel [11] détaillent notamment les relations qui existent entre tous ces indices.

Des indices de dissimilarité sont obtenus à partir de ces indices de similarités par le calcul suivant : $D(x_a, x_b) = 1 - S(x_a, x_b)$. A partir des indices de similarité précédents on obtient alors :

- l'indice de dissimilarité de Jaccard [33] : $D_j(x_a, x_b) = (n_{10} + n_{01}) / (n_{11} + n_{10} + n_{01})$
- l'indice de dissimilarité de Sokal et Sneath [73] : $D_{ss}(x_a, x_b) = ((n_{10} + n_{01}) - (n_{11} + n_{00})) / (n_{10} + n_{01})$
- l'indice de dissimilarité de Russel et Rao [66] : $S_{rr}(x_a, x_b) = (n_{10} + n_{01} + n_{00}) / (n_{11} + n_{10} + n_{01} + n_{00})$

Dans le premier indice de dissimilarité on remarque que les variables valant 0 pour les deux échantillons ne sont pas prises en compte alors que dans le dernier indice ces variables sont prises en compte mais ne comptent pas comme similarité entre les échantillons.

Limite des indices de dissimilarité. Les indices de dissimilarité de Sokal et Sneath et de Russel et Rao ne sont pas des distances mathématiques. L'indice de dissimilarité de Sokal et Sneath n'est pas défini si les vecteurs sont égaux et même en rajoutant la condition que cet indice vaut 0 si les deux vecteurs sont égaux, cet indice est négatif si les valeurs des vecteurs sont égales pour plus de la moitié des variables. L'indice de dissimilarité de Russel et Rao n'est pas non plus une distance car l'indice de deux vecteurs égaux qui ont une valeur égale à 0 pour au moins une variable est strictement positif. Ainsi l'utilisation de ces indices de dissimilarité peut conduire à des incohérences au point de vue des résultats.

L'indice de dissimilarité de Jaccard ne peut pas être utilisé si les vecteurs x_a et x_b sont nuls (si pour toute variable v on a $x_a(v) = x_b(v) = 0$). Par contre, en rajoutant la condition que l'indice de dissimilarité de Jaccard entre deux vecteurs nuls vaut 0 alors l'indice de dissimilarité de Jaccard est bien une distance mathématique. En effet, il est facile de voir qu'il s'agit bien d'une application symétrique positive qui s'annule uniquement si $x_a = x_b$ et Levandowsky et Winter [41] ont prouvé que cet indice vérifie également l'inégalité triangulaire. Pour autant, nous allons voir que la distance de Jaccard n'est pas utilisable pour comparer des chirotopes de modèles en deux ou trois dimensions.

Cette distance est toujours comprise entre 0 et 1 mais dans le cas uniforme elle peut être égale à 1 sans que les vecteurs ne possèdent beaucoup de différences. En effet, prenons l'exemple de deux chirotopes χ_1 et χ_2 tels que pour une base B_1 on ait $\chi_1(B_1) = +$ et $\chi_2(B_1) = -$ et que pour toutes les autres bases B on ait $\chi_1(B) = \chi_2(B) = -$. En transformant χ_1 et χ_2 en vecteurs binaires x_1 et x_2 , on a alors $n_{11}(x_1, x_2) = 0$ et $D_j(x_1, x_2) = 1$.

Nous avons vu que dans le cas non-uniforme, lorsque l'on transforme deux chirotopes χ_1 et χ_2 en vecteurs binaires x_1 et x_2 , chaque $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ correspond à trois variables binaires et dans chaque vecteur binaire une seule de ces trois variables vaut 1. Ainsi si $\chi_1(B) = \chi_2(B)$, x_1 et x_2 ont les mêmes valeurs pour les trois variables binaires correspondant à B , deux de ces valeurs étant égales à 0 et une valeur est égale à 1. Par contre si $\chi_1(B) \neq \chi_2(B)$, parmi les trois variables binaires correspondant à B il y en a une seule qui est égale à zéro pour les deux vecteurs x_1 et x_2 et les deux autres variables ont des valeurs différentes pour x_1 et x_2 (une vaut 1 pour x_1 et 0 pour x_2 et c'est l'inverse pour l'autre variable). Ainsi un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ ayant un signe différent dans les deux chirotopes est comptée deux fois au numérateur et deux fois au dénominateur dans le calcul de la distance de Jaccard, alors qu'un $(d+1)$ -uplet ayant le même signe dans les deux chirotopes est comptée une fois au dénominateur. Ainsi lorsque l'on calcule les distances deux à deux entre plusieurs chirotopes, un $(d+1)$ -uplet non-fixe ne sera pas comptabilisée de la même façon dans les calculs des distances de Jaccard.

5.1.2 Transformation en données numériques

Nous allons maintenant transformer un chirotope d'un modèle en deux ou trois dimensions en un vecteur numérique. La transformation en données numériques a deux motivations : pouvoir utiliser les méthodes de classification qui n'utilisent que ce type de données et pouvoir calculer facilement la somme, la différence ou la moyenne d'un ensemble de vecteurs.

Représentation des chirotopes en données numériques dans le cas uniforme. Dans le cas uniforme la transformation est simple : on remplace le signe + par 1 et le signe - par -1.

En codant les chirotopes de cette façon en vecteurs numériques, la somme de deux vecteurs nous permet de trouver les bases qui ont des signes différents dans les deux chirotopes (la valeur du vecteur somme pour cette base est 0). De même, si l'on prend un ensemble de n chirotopes, le calcul de la somme des n vecteurs permet de trouver les bases fixes pour cet ensemble de chirotopes : une base B sera fixe pour cet ensemble de chirotopes si la valeur du vecteur somme pour la base B est égale à n (dans ce cas le signe de tous les chirotopes est + pour la base B) ou à $-n$ (dans ce cas le signe de tous les chirotopes est - pour la base B).

La moyenne de n vecteurs a également une signification du point de vue des chirotopes. En effet, en notant $n_s(B)$ avec $s \in \{+, -\}$ le nombre de chirotopes dont le signe de la base B est s , la valeur du vecteur moyenne pour la base B est égale à $(n_+(B) - n_-(B))/n$.

Représentation des chirotopes en données numériques dans le cas non-uniforme. Dans le cas non-uniforme, si l'on souhaite avoir des propriétés intéressantes sur la somme et la moyenne de vecteurs, on ne peut pas remplacer simplement le signe + par 1, le signe 0 par le nombre 0 et le signe - par -1. En effet en remplaçant les signes de cette façon, si la valeur du vecteur somme pour

le $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ vaut 0 alors cela pourrait avoir plusieurs causes : soit un chirotope a pour signe + pour B et l'autre a pour signe -, soit les deux chirotopes ont pour signe 0 pour B .

La transformation que nous allons utiliser pour passer d'un chirotope χ à un vecteur numérique v dans le cas non-uniforme est la même que celle pour obtenir un vecteur binaire. Le vecteur numérique sera d'une taille égale à trois fois le nombre de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ et chaque $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ correspondra à trois coordonnées que nous noterons B_- , B_0 et B_+ . Les valeurs du vecteur sont définis de la façon suivante : soit $s \in \{+, 0, -\}$ alors $v(B_s) = 1$ si $s = \chi(B)$ et $v(B_s) = 0$ sinon.

En codant les chirotopes de cette façon en vecteurs numériques, un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ est fixe pour un ensemble de chirotopes si les valeurs du vecteur somme pour deux coordonnées parmi B_- , B_0 et B_+ est égale à 0 et le signe de tous les chirotopes correspond à la coordonnée non nulle.

Concernant la moyenne de n vecteurs, la valeur du vecteur moyenne pour la coordonnée B_s est égale à la proportion de chirotopes ayant pour signe s pour le $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$.

Comparaisons de données numériques. La distance la plus fréquemment utilisée pour comparer deux vecteurs numériques x_a et x_b est la distance euclidienne ou distance L_2 que l'on notera d_2 (on verra par la suite d'où provient le "2"). En notant l la taille des vecteurs, la distance L_2 se définit de la façon suivante :

distance euclidienne

$$d_2(x_a, x_b) = \sqrt{\sum_{j=1}^l (x_a(j) - x_b(j))^2}.$$

Pour des points en deux dimensions cette distance correspond à la distance en ligne droite dans le plan.

Une autre distance souvent utilisée est la distance de Manhattan ou distance L_1 (que l'on notera d_1). La formule est la suivante :

distance de Manhattan

$$d_1(x_a, x_b) = \sum_{j=1}^l |x_a(j) - x_b(j)|.$$

Cette distance s'appelle la distance de Manhattan puisqu'elle correspond à la distance parcourue pour aller d'un point à un autre dans une ville comme Manhattan, dans laquelle les bâtiments empêchent de se déplacer en ligne droite et les rues sont perpendiculaires les unes aux autres. Cette distance se différencie de la précédente par le fait que les valeurs extrêmes sont atténuées par le fait qu'elles ne sont pas élevées au carré.

Une troisième distance existe mais est plus rarement utilisée : la distance de Tchebychev ou distance L_∞ (que l'on notera d_∞). La formule est la suivante :

$$d_\infty(x_a, x_b) = \max_{1 \leq j \leq l} |x_a(j) - x_b(j)|.$$

Ces trois distances sont des cas particuliers de la distance de Minkowski dont la formule est la suivante :

$$d_p(x_a, x_b) = \sqrt[p]{\sum_{j=1}^l (x_a(j) - x_b(j))^p}.$$

A partir de cette distance on obtient les distances L_1 , L_2 et L_∞ en remplaçant p respectivement par 1, 2 et l'infini (ce qui correspond à un passage à la limite et consiste donc à prendre la valeur maximale).

Parmi les distances précédentes, seule la distance L_1 a une signification du point de vue des chirotopes. En effet, dans le cas uniforme les coordonnées des vecteurs sont soit -1 soit 1 . Ainsi la distance L_1 est égale à deux fois le nombre de coordonnées différentes et donc à deux fois la distance entre deux chirotopes définie à la Section 2.3.

Dans le cas non-uniforme les coordonnées des vecteurs sont soit 0 soit 1 . La distance L_1 compte le nombre de coordonnées différentes entre deux vecteurs. Or dans le cas non-uniforme, chaque $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ correspond à trois coordonnées. Si deux chirotopes ont des signes différents pour le $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ alors parmi les trois coordonnées correspondant à B , les vecteurs numériques correspondant à ces chirotopes diffèrent pour deux coordonnées. Comme pour le cas uniforme, la distance L_1 est égale à deux fois la distance entre deux chirotopes dans le cas non-uniforme.

Représentation des chirotopes par des points. Pour illustrer certaines méthodes de classification (comme la méthode des k-means par exemple) nous aurons besoin de représenter les chirotopes par des points. En notant par l le nombre de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$, nous considérerons alors que les vecteurs numériques obtenus précédemment sont des points de $[-1, 1]^l$ dans le cas uniforme et de $[0, 1]^{3 \times l}$ dans le cas non-uniforme.

Nous serons amenés à utiliser la notion de point moyen d'un ensemble de points. Le *point moyen d'un ensemble de points* $P = \{p_1, \dots, p_n\}$ est le point m pour lequel chaque coordonnée i est la moyenne des coordonnées i des points de P . Formellement :

$$m(i) = \frac{\sum_{j=1}^n p_j(i)}{n}$$

Soit $\mathcal{C} = \{\chi_1, \chi_2, \dots, \chi_n\}$ un ensemble de chirotopes. On note p_i le point représentant χ_i . Par abus de langage, on dira que le point m est le point moyen de \mathcal{C} si m est le point moyen de l'ensemble de points $\{p_1, p_2, \dots, p_n\}$.

Propriété 5.1. *Les coordonnées du point moyen d'un ensemble de chirotopes \mathcal{C} nous permettent de déterminer la proportion de chirotopes de \mathcal{C} ayant un signe donné pour chaque $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$.*

En effet, dans le cas uniforme, les chirotopes sont transformés en points de $[-1, 1]^l$ où $l = \binom{n}{d+1}$. Ainsi chaque coordonnée correspond au signe d'une base de $\mathcal{B}(\mathcal{E})$. En notant m le point moyen de l'ensemble de chirotopes \mathcal{C} , on a alors que pour toute base B de $\mathcal{B}(\mathcal{E})$, $m(B)$ est la moyenne des $p_j(B)$ avec $p_j \in \mathcal{C}$. Ainsi $m(B) = 1$ (respectivement -1) si et seulement si pour tout $p_j \in \mathcal{C}$ on a $p_j(B) = 1$ (resp. -1), c'est-à-dire que tous les chirotopes classés dans \mathcal{C} ont comme signe $+$ (resp. $-$) pour la base B . De même, $m(B) \in]0, 1[$ (respectivement $] -1, 0[$) si et seulement si la majorité des chirotopes classés dans \mathcal{C} ont comme signe $+$ (resp. $-$) pour la base B . Plus précisément on a $m(B) = (n_+(B) - n_-(B))/n$ où n est le nombre de chirotopes de \mathcal{C} et $n_+(B)$ (respectivement $n_-(B)$) est le nombre de chirotopes de \mathcal{C} ayant comme signe $+$ (resp. $-$) pour la base B . Ainsi $m(B) = (n -$

$2 \times n_-(B)/n$ ce qui permet de calculer à partir du point moyen de \mathcal{C} la proportion de chirotopes de \mathcal{C} ayant un signe donnée pour chaque base B .

Dans le cas non-uniforme, chaque chirotope est transformé en point de $[0, 1]^{3 \times l}$ et chaque $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ est associée à trois coordonnées correspondant aux trois signes. Ces coordonnées sont notées B_+ , B_0 et B_- . Ainsi pour toute coordonnée B_s où B est un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ et $s \in \{+, 0, -\}$, on a $m(B_s) = 1$ (respectivement 0) si et seulement si tout (resp. aucun) chirotope de \mathcal{C} a comme signe s pour le $(d+1)$ -uplet B . De même, $m(B_s) \in]1/2, 1[$ si et seulement si la majorité des chirotopes classés dans \mathcal{C} ont comme signe s pour le $(d+1)$ -uplet B . Plus précisément on a $m(B_s) = n_s(B)/n$ où n est le nombre de chirotopes de \mathcal{C} et $n_s(B)$ est le nombre de chirotopes de \mathcal{C} ayant comme signe s pour le $(d+1)$ -uplet B . Ainsi les coordonnées du point moyen de \mathcal{C} sont exactement les proportions de chirotopes de \mathcal{C} ayant un signe donné pour chaque $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$.

5.2 Méthodes de clustering

Les méthodes de clustering peuvent être séparées en deux grandes familles : les méthodes hiérarchiques et les méthodes de partitionnement. Les méthodes hiérarchiques créent un système de clusters emboîtées, de sorte que plus les clusters sont petits et plus ils correspondent à des éléments qui sont semblables. Les méthodes de partitionnement créent des clusters qui forment une partition de l'ensemble des échantillons, c'est-à-dire que chaque échantillon appartient à un unique cluster et donc les clusters sont deux à deux disjoints. Pour plus de détails sur ces méthodes, nous renvoyons le lecteur à [49] et [78].

Comme il existe un grand nombre de méthodes, nous n'étudions dans cette section que quelques méthodes classiques. Nous commençons par étudier la *classification hiérarchique* puis nous expliquons pourquoi nous n'avons pas appliqué cette méthode. Nous étudions ensuite la méthode des *k-means* et nous montrons qu'elle est facilement utilisable pour classer des chirotopes et donne des résultats satisfaisants. Bien que n'étant pas combinatoire, nous utiliserons cette méthode pour classer des chirotopes (voir Partie V). Nous nous intéressons ensuite à la méthode des *k-medoids* qui est proche de la méthode des *k-means* et qui à l'avantage d'être combinatoire. Les calculs étant plus rapides que dans l'algorithme des *k-means* mais la classification obtenue étant parfois moins bonne, nous utiliserons généralement les *k-medoids* pour déterminer rapidement si notre méthode basée sur les matroïdes orientés peut s'appliquer à un jeu de données et s'il est donc utile de calculer une classification à l'aide de l'algorithme des *k-means*. Nous évoquons enfin quelques autres méthodes de clustering et nous expliquons pourquoi nous ne les avons pas utilisées.

Dans la suite de ce chapitre nous utiliserons la notion d'arbre qui provient de la théorie des graphes. Un *arbre* est un ensemble non vide et fini d'éléments appelés *nœuds* et liés par une relation de parenté. Un père est relié avec chacun de ses fils par une *arête*. Dans un arbre il existe un unique nœud n'ayant pas de père, appelé *racine* de l'arbre. Les *feuilles* sont les éléments ne possédant pas de fils. Un élément qui n'est pas une feuille est appelé *nœud interne*. La *hauteur* d'un arbre est le nombre maximum d'arêtes entre la racine et une feuille.

arbre

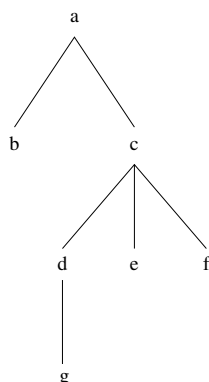


FIGURE 5.1 – Exemple d'arbre

Exemple. La Figure 5.1 représente un arbre contenant 7 nœuds numérotés de a à g . Le nœud a est le père des nœuds b et c , les nœuds d , e et f sont les fils du nœud c et le nœud g est le fils du nœud d . La racine de cet arbre correspond au nœud a et les feuilles sont les nœuds b , e , f et g . La hauteur de cet arbre vaut 3.

5.2.1 Classification ascendante hiérarchique

Présentation de la méthode

Le but de la classification ascendante hiérarchique est de faire apparaître la hiérarchie qu'il y a au sein de la classification. Une hiérarchie est une notion mathématique définissant les relations d'inclusion entre des parties d'un même ensemble. Les parties d'une hiérarchie sont soit disjointes soit incluses l'une dans l'autre. Formellement, soient E un ensemble d'éléments $\{e_1, e_2, \dots, e_n\}$ et $\mathcal{H} = \{H_1, H_2, \dots, H_k\}$ un ensemble de parties de E . \mathcal{H} est une *hiérarchie* si et seulement si :

- $E \in \mathcal{H}$
- $\forall i \in \{1, \dots, n\}, \{e_i\} \in \mathcal{H}$
- $\forall i, j \in \{1, \dots, k\}$, on a soit $H_i \subset H_j$, soit $H_j \subset H_i$, soit $H_i \cap H_j = \emptyset$

Exemple. Pour $E = \{e_1, e_2, e_3, e_4, e_5\}$, $\mathcal{H} = \{\{e_1\}, \{e_2\}, \{e_3\}, \{e_4\}, \{e_5\}, \{e_1, e_2\}, \{e_4, e_5\}, \{e_3, e_4, e_5\}, E\}$ est une hiérarchie.

On appelle *hiérarchie indexée* le couple (\mathcal{H}, f) où \mathcal{H} est une hiérarchie et f est une fonction de \mathcal{H} dans \mathbb{R}^+ vérifiant :

- $\forall e_i \in E, f(\{e_i\}) = 0$;
- $\forall i, j \in \{1, \dots, k\}$ tels que $H_i \subsetneq H_j$, on a $f(H_i) < f(H_j)$.

On appelle f *fonction d'indexation*.

Exemple. On associe à la hiérarchie de l'exemple précédent à la fonction f telle que :

- $\forall e_i \in E, f(\{e_i\}) = 0$;
- $f(\{e_1, e_2\}) = 1,4$, $f(\{e_4, e_5\}) = 0,8$;

- $f(\{e_3, e_4, e_5\}) = 2, 1, \quad f(E) = 3, 6.$

On utilise un *arbre hiérarchique* (ou *dendrogramme*) pour représenter une hiérarchie. Dans cet arbre chaque nœud correspond à une partie de la hiérarchie. La racine de l'arbre correspond à l'ensemble E et chaque feuille correspond à un singleton (i.e. partie qui ne contient qu'un seul élément de E). Il existe une arête dans l'arbre entre deux parties de la hiérarchie si et seulement si une de ces deux parties est incluse dans l'autre. Si la hiérarchie est indexée, on représente la valeur de la fonction d'indexation sur l'axe des abscisses. La Figure 5.2 représente un arbre hiérarchique correspondant à la hiérarchie non-indexée de l'exemple. Les nœuds internes sont représentés par des ronds noirs. La Figure 5.3 représente la hiérarchie indexée de l'exemple. A noter que la représentation de l'arbre hiérarchique n'est pas unique si la hiérarchie n'est pas indexée. Dans la Figure 5.2, le nœud correspondant à la partie $\{e_1, e_2\}$ est représenté plus proche de la racine que le nœud correspondant au cluster $\{e_3, e_4, e_5\}$. On aurait pu représenter le nœud correspondant à la partie $\{e_1, e_2\}$ comme étant le nœud interne le plus éloigné de la racine. Pour une hiérarchie indexée, comme l'axe des abscisses représente la fonction d'indexation, la représentation est unique (à réordonnement des éléments de E près).

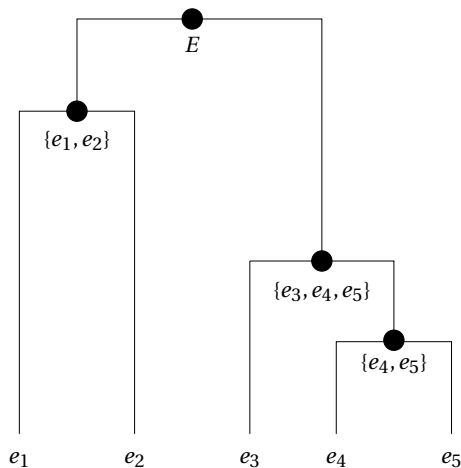


FIGURE 5.2 – Arbre hiérarchique correspondant à la hiérarchie de l'exemple

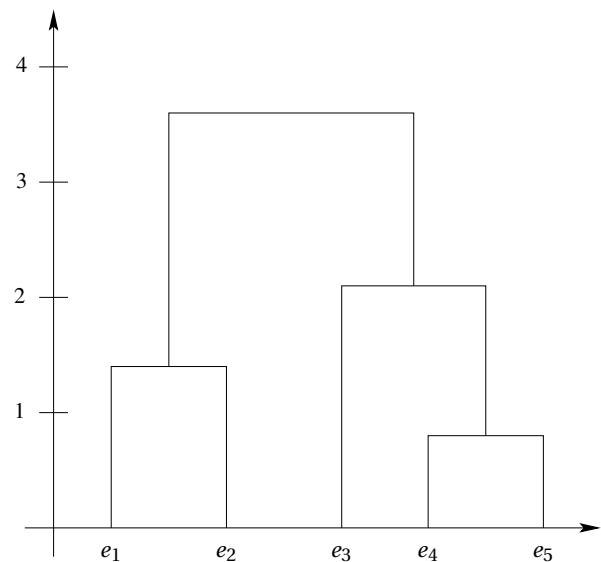


FIGURE 5.3 – Arbre hiérarchique représentant la hiérarchie indexée de l'exemple

Pour répondre au problème de classification, on utilise des hiérarchies indexées dans lesquelles la fonction d'indexation représente une distance entre les parties de la hiérarchie. Une fois que l'on a créé l'arbre hiérarchique, on choisit à quel endroit nous allons couper l'arbre par une droite horizontale afin d'obtenir les clusters. On peut couper l'arbre hiérarchique d'au plus n façons différentes. En effet, chaque nœud interne de l'arbre correspond à une séparation d'une partie de E en au moins deux sous-parties. L'arbre a alors au plus $n - 1$ nœuds internes. Si la fonction d'indexation a une valeur différente pour chaque nœud interne alors il y a au plus $n + 1$ façons différentes de couper l'arbre. En ordonnant les nœuds internes par rapport à la valeur de la fonction d'indexation

par ordre croissant, on peut alors couper entre les feuilles et le premier nœud interne ou entre le premier et le deuxième nœud interne et ainsi de suite jusqu'à couper entre l'avant dernier nœud interne et la racine. La dernière coupe possible est de couper au dessus de la racine.

Chacune de ces coupes correspond à une classification différente. Si la coupe a lieu entre les feuilles et le premier nœud interne cela correspond à avoir n clusters contenant chacun un élément distinct de E . A l'inverse si la coupe a lieu au dessus de la racine alors on obtient un seul cluster qui correspond à l'ensemble E en entier. A part si la coupe a lieu au dessus de la racine, lorsque l'on coupe l'arbre selon une droite horizontale, cette droite croise un certain nombre d'arêtes. La classification correspondante à cette coupe a un nombre de clusters égal au nombre d'arêtes croisées par la droite.

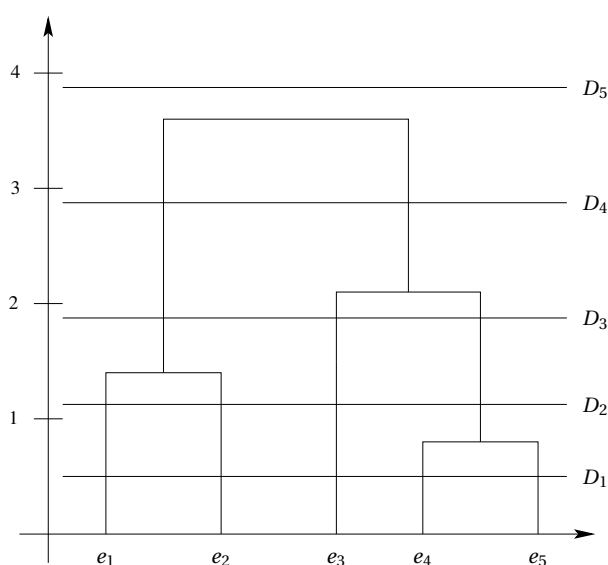


FIGURE 5.4 – Arbre hiérarchique correspondant à la hiérarchie de l'exemple

Exemple. La Figure 5.4 représente les différentes coupes que l'on peut faire sur l'arbre hiérarchique de l'exemple précédent. Le Tableau 5.1 indique les clusters obtenus en fonction de la droite nous servant à couper.

droite	cluster(s)
D_1	$\{e_1\}, \{e_2\}, \{e_3\}, \{e_4\}, \{e_5\}$
D_2	$\{e_1\}, \{e_2\}, \{e_3\}, \{e_4, e_5\}$
D_3	$\{e_1, e_2\}, \{e_3\}, \{e_4, e_5\}$
D_4	$\{e_1, e_2\}, \{e_3, e_4, e_5\}$
D_5	$\{e_1, e_2, e_3, e_4, e_5\}$

TABLE 5.1 – Clusters en fonction de la droite choisie dans la Figure 5.4

Si l'utilisateur connaît à priori le nombre de clusters et qu'il existe une droite horizontale croisant ce nombre d'arêtes alors on obtient facilement les clusters. Si tel n'est pas le cas il faut alors étudier la fonction d'indexation. A partir de la fonction d'indexation il y a plusieurs façons de déterminer une coupe. En ordonnant nœuds internes de l'arbre par rapport à la valeur de la fonction d'indexation par ordre croissant, on peut par exemple couper entre deux nœuds internes consécutifs x_i et x_{i+1} tels que $f(x_{i+1}) - f(x_i)$ soit maximal. Une autre solution pour déterminer la coupe est de définir un seuil au-delà duquel les nœuds internes de l'arbre correspondent à des fusions de parties qui sont trop différentes. Ainsi le choix de la fonction d'indexation est très important car cela peut donner des résultats très différents sur les clusters retenus après la coupe.

Il y a plusieurs façons de définir une distance entre des ensembles. La plupart des méthodes se basent sur la matrice de distances des éléments de E . Parmi les méthodes les plus utilisées pour calculer la distance entre deux ensembles, il y a ces trois distances :

- la distance maximale qui consiste à prendre la plus grande distance entre toutes les paires d'éléments appartenant à des ensembles différents. Formellement on a

$$d(A, B) = \max\{d(a, b) | a \in A, b \in B\};$$

- la distance minimale qui consiste à prendre la plus petite distance entre toutes les paires d'éléments appartenant à des ensembles différents. Formellement on a

$$d(A, B) = \min\{d(a, b) | a \in A, b \in B\};$$

- la distance moyenne qui consiste à prendre la moyenne de toutes les distances entre toutes les paires d'éléments appartenant à des ensembles différents. Formellement on a

$$d(A, B) = \sum_{a \in A} \sum_{b \in B} d(a, b) / (|A| \times |B|).$$

Il existe d'autres façons de définir des distances entre des ensembles comme par exemple la méthode de Ward [84] qui consiste à minimiser l'inertie intraclasse.

Il est important de noter que nous avons utilisé le terme de distance entre deux ensembles A et B pour les fonctions précédentes alors qu'aucune de ces fonctions n'est une distance au sens mathématique du terme.

Exemple. Nous allons illustrer les différentes distances par un exemple dans lequel on considère un ensemble de sept échantillons. La Table 5.2 indique les distances entre toutes les paires d'échantillons.

	a	b	c	d	e	f	g
a	0						
b	45	0					
c	8	38	0				
d	36	10	29	0			
e	41	4	34	7	0		
f	40	9	36	11	6	0	
g	32	14	24	9	10	8	0

TABLE 5.2 – Distances entre 7 échantillons

La Figure 5.5 représente les arbres hiérarchiques obtenus selon que l'on utilise la distance maximale, la distance minimale ou la distance moyenne. On remarque que les trois hiérarchies sont différentes. Ainsi si l'on souhaite obtenir quatre clusters, en utilisant la distance maximale on obtiendrait les clusters $\{b, e\}$, $\{d\}$, $\{g, f\}$ et $\{c, a\}$, alors qu'avec la distance moyenne on obtiendrait les clusters $\{b, e, f\}$, $\{d\}$, $\{g\}$ et $\{c, a\}$.

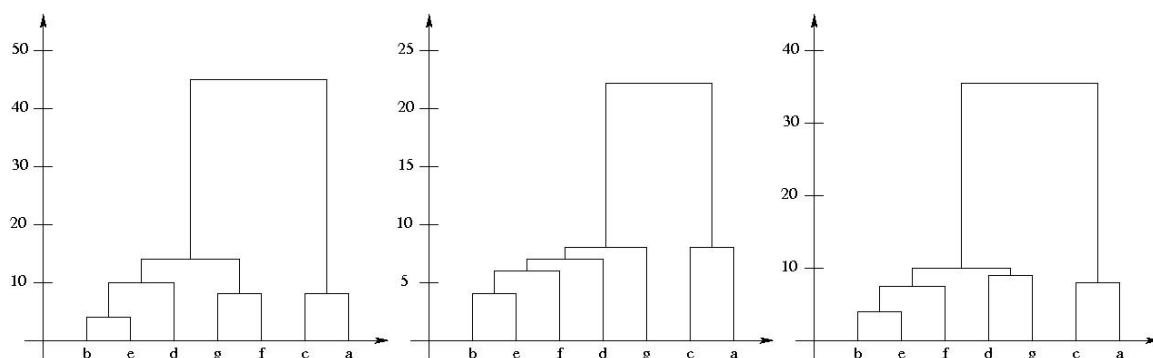


FIGURE 5.5 – Arbres hiérarchiques correspondant aux échantillons de la Table 5.2 en fonction de la distance utilisée. A gauche l'arbre correspond à l'utilisation de la distance maximale, au milieu l'arbre correspond à la distance minimale et à droite l'arbre correspond à l'utilisation de la distance moyenne. A noter que l'ordre des échantillons sur l'axe horizontal est différent entre l'arbre de gauche et les deux autres arbres.

Nous allons voir que cette méthode de clustering a une complexité polynomiale. Le premier point est de calculer la matrice de distance entre tous les échantillons. Quelle que soit la distance utilisée, le calcul de la distance entre deux échantillon nécessite $O(l)$ opérations où l est la taille des vecteurs représentant chaque échantillon. Ainsi le calcul de la matrice de distance se fait en $O(n^2 \times l)$ opérations.

La construction de l'arbre se déroule en $n - 1$ étapes qui correspondent chacune à un nœud interne de l'arbre. Pour construire le nœud interne faisant passer le nombre de clusters de $k + 1$ à k , on doit prendre le minimum des distances entre toutes les paires de clusters ce qui revient à parcourir un tableau de taille $(k + 1) \times k/2$, soit effectuer $O(k^2)$ opérations. Après la création du nœud interne, il faut mettre à jour les distances entre les clusters. Ce nœud interne correspond à un nouveau cluster qui est le résultat de la fusion de deux anciens clusters. Notons \mathcal{C}_1 ce nouveau cluster et $\mathcal{C}_2, \dots, \mathcal{C}_k$ les clusters qui ne changent pas. Il y a alors $k - 1$ distances à calculer puisque les distances ne faisant pas intervenir le nouveau cluster ne changent pas. Ainsi, pour tout cluster \mathcal{C}_j avec $2 \leq j \leq k$, il faut calculer $d(\mathcal{C}_1, \mathcal{C}_j)$ ce qui nécessite $O(l \times |\mathcal{C}_1| \times |\mathcal{C}_j|)$ calculs. Or pour tout k on a $\sum_{j=1}^k |\mathcal{C}_j| = n$. Ainsi la mise à jour des distances entre les clusters se fait en $O(l \times |\mathcal{C}_1| \times n)$.

Ainsi la construction de l'arbre s'effectue en $O(l \times n^3)$ où l est la taille des vecteurs représentant chaque échantillon.

L'algorithme que nous venons de voir est l'algorithme le plus simple pour réaliser une classification ascendante hiérarchique. Des algorithmes plus perfectionnés permettent d'obtenir une complexité en $O(l \times n^2)$ pour construire l'arbre hiérarchique. D'autres méthodes de classification

ascendante hiérarchique existent également. Certaines se basent sur des critères de densité alors que d'autres cherchent des classes de forme arbitraire. Toutes ces méthodes et algorithmes sont notamment détaillés dans [49].

A noter qu'il existe une autre sorte de classification hiérarchique qui est la classification hiérarchique descendante. Cette méthode consiste à créer la hiérarchie à partir d'un unique cluster qui est formé par l'ensemble E en entier, puis de diviser les clusters jusqu'à obtenir n clusters qui sont les singletons des éléments de E . Cette méthode est moins utilisée en pratique puisque plus gourmande en calculs. En effet, il existe 2^n façons différentes de séparer un cluster en deux, ce qui fait qu'à chaque étape la méthode descendante ne fait pas un nombre de calculs polynomial mais exponentiel.

Avantages et inconvénients pour la classification de chirotopes

On peut facilement appliquer cette méthode pour classer des chirotopes puisque cette méthode se base uniquement sur une matrice de distances. De plus cette méthode possède un gros avantage puisqu'à partir de l'arbre hiérarchique on peut voir facilement comment est formé chaque cluster et repérer d'éventuels éléments atypiques. Il y a en effet deux façons de former un cluster, soit ce cluster est formé par fusion de deux sous-clusters soit il est formé en rajoutant un élément à un sous-cluster. Dans le premier cas il peut être intéressant d'étudier ces deux sous-clusters alors que lorsqu'un élément est rajouté tardivement à un cluster, il y a de fortes chances que cet élément soit atypique.

Par contre cette méthode présente trois inconvénients. Le premier inconvénient réside dans le choix de la coupe. En effet, pour obtenir une partition en clusters à partir d'un arbre hiérarchique il faut choisir où couper l'arbre. Si l'on connaît par avance le nombre de clusters que l'on souhaite obtenir on sait alors directement où couper l'arbre, mais il arrive que l'on ne connaisse pas à l'avance ce nombre.

Le deuxième inconvénient de la méthode est l'influence de la distance interclasse sur les résultats. En effet, cette méthode nécessite l'utilisation d'une distance interclasse. Or nous avons vu dans la Figure 5.5 qu'en utilisant différentes distances interclasses sur un même jeu de données, on arrivait à des arbres hiérarchiques très différents conduisant à des clusters différents. Ainsi pour un même jeu de données cette méthode peut donner des résultats différents selon la distance interclasse utilisée.

Le troisième inconvénient de cette méthode est qu'elle ne permet pas de caractériser les clusters par un sous-ensemble de $d + 1$ -uplets. Dans cette méthode on regroupe les chirotopes ayant les distances les plus petites en ne se basant que sur la matrice des distances. Ainsi la distance est globale et rien ne permet d'avoir des informations sur les signes des $d + 1$ -uplets des chirotopes au sein de chaque cluster.

Ces inconvénients nous ont conduits à ne pas utiliser cette méthode.

5.2.2 k -means

Dans cette section nous allons présenter la méthode de partitionnement des k -means. Nous utiliserons principalement cette méthode pour classer les chirotopes lors de nos applications (voir Partie V).

diagramme
de Voronoï

Dans la suite nous serons amenés à parler du diagramme de Voronoï. Le *diagramme de Voronoï* de \mathbb{R}^l en fonction d'un ensemble S de r éléments f_1, \dots, f_r de \mathbb{R}^l est une décomposition de l'espace en r régions. Chaque région est associée à un élément, de sorte que la région associée à f_i est constituée de l'ensemble des points de \mathbb{R}^l qui sont plus proches de f_i que de tout autre élément de S .

Présentation de la méthode

k -means

La méthode des *k-means* est utilisée pour partitionner des données numériques en un nombre k de clusters fixé à l'avance. Chaque échantillon est considéré comme étant un point de \mathbb{R}^l et la méthode utilise comme distance la distance euclidienne. Le but est alors de trouver une partition des points en k clusters de sorte que cette partition minimise la somme de la dispersion des clusters. La dispersion d'un cluster est calculée par l'effectif du cluster multiplié par la variance du cluster. En d'autres termes, on minimise la somme pour chaque cluster des distances au carré entre le point moyen du cluster et les points de ce cluster. Formellement, en notant E l'ensemble des échantillons e_i , $\mathcal{C}_1, \dots, \mathcal{C}_k$ les clusters et c_1, \dots, c_k leur point moyen, on cherche le minimum global de la fonction suivante :

$$f(E, \mathcal{C}_1, \dots, \mathcal{C}_k, c_1, \dots, c_k) = \sum_{j=1}^k \sum_{e_i \in \mathcal{C}_j} (d_2(e_i, c_j))^2$$

Dans ce calcul, on utilise le carré des distances notamment pour éviter d'avoir dans un cluster des points (même en petit nombre) qui soient très éloignés du centre du cluster.

Le problème a été largement étudié dans la littérature et a été prouvé comme étant *NP-complet* si l'on ne connaît pas le nombre k de clusters à chercher [2]. Il a été montré que la solution du problème des *k-means* est égale au diagramme de Voronoï des points moyens des clusters [83]. Par contre si la valeur de k est fixée, un algorithme polynomial en temps $O(n^{lk+1})$ a été proposé pour résoudre le problème [32]. Cet algorithme consiste à tester tous les diagrammes de Voronoï possibles des n échantillons en k régions. Par contre, cet algorithme ne peut pas être utilisé en pratique car les valeurs de n et/ou l peuvent être très grandes. Ainsi l'heuristique qui est utilisée le plus souvent se base sur les trois étapes suivantes :

- **Initialisation.** Choisir aléatoirement k points c_1, \dots, c_k de \mathbb{R}^p et créer k clusters vides $\mathcal{C}_1, \dots, \mathcal{C}_k$.
- **Mise à jour des clusters.** Pour chaque échantillon e_i , déterminer le point c_j le plus proche de e_i (ou en choisir un si le point n'est pas unique) et affecter e_i au cluster \mathcal{C}_j . Formellement :

$$\mathcal{C}_j = \{e_i \mid d(e_i, c_j) = \min_l d(e_i, c_l)\}$$
 avec e_i appartiennent exactement à un cluster.
- **Mise à jour des points moyens.** Pour chaque cluster \mathcal{C}_j calculer le point moyen c_j de \mathcal{C}_j . Formellement :

$$c_j = (\sum_{e_i \in \mathcal{C}_j} e_i) / |\mathcal{C}_j|.$$

Une fois que l'étape d'initialisation est réalisée, l'algorithme répète les étapes de mise à jour des clusters et des points moyens jusqu'à ce qu'il n'y ait plus de changement.

Nous allons voir que l'algorithme converge. Notons $\mathcal{C}_j^{(t)}$ le cluster \mathcal{C}_j après t étapes de mise à jour des clusters et $c_j^{(t)}$ son point moyen après t étapes de mise à jour des points moyens. Nous allons commencer par étudier la variation de la fonction f au cours de l'étape $t+1$ de mise à jour

des clusters. Si aucun échantillon ne change de cluster alors la valeur de la fonction ne change pas au cours de l'étape. Nous allons supposer qu'un échantillon e_i change de cluster. Ainsi, sans perdre de généralités on peut supposer que l'échantillon e_i appartient au cluster $\mathcal{C}_1^{(t)}$ avant la mise à jour et au cluster $\mathcal{C}_2^{(t+1)}$ après la mise à jour. On a alors $d(e_i, c_2^{(t+1)}) \leq d(e_i, c_1^{(t)})$. On a donc

$$f(E, \mathcal{C}_1^{(t+1)}, \dots, \mathcal{C}_k^{(t+1)}, c_1^{(t)}, \dots, c_k^{(t)}) \leq f(E, \mathcal{C}_1^{(t)}, \dots, \mathcal{C}_k^{(t)}, c_1^{(t)}, \dots, c_k^{(t)}).$$

Etudions maintenant la variation de la fonction f au cours de l'étape $t + 1$ de mise à jour des points moyens. Nous allons montrer que le point moyen d'un ensemble de points minimise la somme des carrés des distances. Soient $\{p_1, \dots, p_n\}$ un ensemble de points de \mathbb{R}^d et c le point moyen de cet ensemble de points. On a alors $\sum_{i=1}^n \overrightarrow{cp_i} = \vec{0}$. On cherche le point $q \in \mathbb{R}^d$ minimisant $\sum_{i=1}^n \|\overrightarrow{qp_i}\|^2$. Or en notant $\vec{u} \times \vec{v}$ le produit scalaire entre \vec{u} et \vec{v} on a

$$\begin{aligned} \sum_{i=1}^n \|\overrightarrow{qp_i}\|^2 &= \sum_{i=1}^n \|\overrightarrow{qc} + \overrightarrow{cp_i}\|^2 \\ &= \sum_{i=1}^n (\|\overrightarrow{qc}\|^2 + 2\overrightarrow{qc} \times \overrightarrow{cp_i} + \|\overrightarrow{cp_i}\|^2) \\ &= n\|\overrightarrow{qc}\|^2 + 2\overrightarrow{qc} \times \left(\sum_{i=1}^n \overrightarrow{cp_i}\right) + \sum_{i=1}^n \|\overrightarrow{cp_i}\|^2 \end{aligned}$$

car le produit scalaire est symétrique et bilinéaire. Comme $\sum_{i=1}^n \overrightarrow{cp_i} = \vec{0}$ le deuxième terme de la somme est nul et comme le troisième est une constante la somme est minimale pour $q = c$. Ainsi le point moyen d'un ensemble de points minimise la somme des carrés des distances d'où $\sum_{e_i \in \mathcal{C}_j} d^2(e_i, c_j^{(t+1)}) \leq \sum_{e_i \in \mathcal{C}_j} d^2(e_i, c_j^{(t)})$ pour tout cluster \mathcal{C}_j . On a donc

$$f(E, \mathcal{C}_1^{(t+1)}, \dots, \mathcal{C}_k^{(t+1)}, c_1^{(t+1)}, \dots, c_k^{(t+1)}) \leq f(E, \mathcal{C}_1^{(t)}, \dots, \mathcal{C}_k^{(t)}, c_1^{(t)}, \dots, c_k^{(t)}).$$

L'algorithme converge vers un minimum local qui dépend de l'étape d'initialisation. Cette étape est équivalente à choisir aléatoirement une partition de l'ensemble des échantillons en k clusters. Comme il y a une infinité de choix pour les points c_1, \dots, c_k dans l'étape d'initialisation et de l'ordre de k^n choix si l'on décide d'initialiser par une partition, il est courant que l'étape d'initialisation consiste à choisir aléatoirement k points c_1, \dots, c_k parmi l'ensemble des échantillons. Par contre, rien ne nous garantit qu'une telle initialisation conduise à une solution correspondant à un minimum global.

Nous allons étudier les temps de calcul de cet algorithme. Chaque étape de mise à jour des clusters s'effectue en $O(n \times l \times k)$ opérations (où l est la taille des vecteurs) puisque pour chacun des n échantillons il faut calculer la distance entre cet échantillon et le point c_j pour $1 \leq j \leq k$. Pour chaque cluster \mathcal{C}_j , l'étape de mise à jour de son point moyen requiert $O(|\mathcal{C}_j| \times l)$ opérations. Ainsi chaque étape de mise à jour des points moyens se fait en $O(\max |\mathcal{C}_j| \times l \times k)$. La complexité de cet algorithme est donc de $O(n \times l \times k \times I)$ où I est le nombre d'itérations, mais en général le nombre d'itération nécessaire pour obtenir la convergence est petit (< 10) [15]. On peut alors compenser le fait que l'algorithme ne retourne qu'une solution correspondant à un minimum local en exécutant plusieurs fois l'algorithme avec des initialisations différentes.

De nombreuses améliorations ont été apportées à cette méthode que ce soit entre autres pour améliorer la classification [3, 28, 43, 87] ou pour trouver automatiquement le nombre k de clusters [56, 29].

Lorsque l'on calcule la distance entre deux points, pour chaque coordonnée, la distance euclidienne nécessite de calculer le carré de la différence entre les coordonnées alors que pour la distance de Manhattan on ne calcule que la valeur absolue de cette différence. Ainsi pour chaque coordonnée, le calcul de la distance de Manhattan est plus rapide que la distance euclidienne. Une étude a été faite pour comparer le résultats de l'algorithme si on utilise comme distance la distance euclidienne ou la distance de Manhattan [44]. Celle-ci montre que sur le jeu de donnée utilisé, pour la plupart des valeurs de k comprises entre 2 et 25, l'utilisation de la distance de Manhattan permet d'obtenir la convergence avec un nombre d'itérations plus petit que la distance euclidienne. Comme la distance de Manhattan ne nécessite que des additions, contrairement à distance euclidienne qui nécessite des multiplications, le temps de calcul pour atteindre la convergence avec la distance de Manhattan est alors réduit. L'étude montre également que le changement de la distance peut modifier la classification des données.

Avantages et inconvénients pour la classification de chirotopes

On peut facilement utiliser cette méthode pour faire une classification d'un ensemble de chirotopes en transformant les chirotopes en vecteurs numériques (voir Section 5.1.2). De plus nous venons de voir que cette méthode permet d'utiliser la distance de Manhattan comme distances entre deux échantillons, ce qui correspond à un facteur multiplicatif près à la distance que nous avons définie sur les chirotopes. Nos données étant codées par des vecteurs numériques de grande dimension (nous avons près de 2 700 000 bases non-fixes pour les applications comptant le plus de bases (voir Chapitre 11), l'utilisation de la distance de Manhattan permet également de diminuer le temps de calcul.

Cette méthode présente en plus deux avantages. Le premier avantage est qu'en calculant la distance entre chaque chirotope et le point moyen du cluster auquel il appartient, on obtient très facilement pour chaque cluster les chirotopes qui sont le plus au centre du cluster et les chirotopes les plus éloignés du centre du cluster. La détection de ces derniers est une étape importante puisque les chirotopes les plus éloignés des centres peuvent être des échantillons atypiques.

Le deuxième avantage est que pour chaque cluster \mathcal{C}_i , on peut obtenir une caractérisation de \mathcal{C}_i en utilisant les coordonnées du point moyen de \mathcal{C}_i . En effet, d'après la Propriété 5.1, le point moyen de chaque cluster \mathcal{C}_i nous donne la proportion de chirotopes de \mathcal{C}_i ayant un signe donné pour chaque $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$. On peut ainsi déterminer simplement s'il existe des $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ nous permettant de discriminer les classes, c'est-à-dire que l'on peut déterminer pour chaque cluster \mathcal{C}_i s'il existe un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ et un signe s tel que le chirotope χ appartient à \mathcal{C}_i si et seulement si le $\chi(B) = s$.

Cette méthode a un léger inconvénient pour classer des chirotopes puisqu'elle n'est pas combinatoire. En effet, nous avons codé les formes par des chirotopes c'est-à-dire des vecteurs dont les coordonnées ont un nombre fini de valeurs possibles (+, 0 ou -). Par contre cette méthode nécessite d'utiliser des points moyens dont les coordonnées peuvent être n'importe quelle valeur de $[-1, 1]$ dans le cas uniforme ou n'importe quelle valeur de $[0, 1]$ dans le cas non-uniforme.

Cette méthode présentant deux avantages très intéressants, nous avons décidé d'utiliser la méthode des k -means pour obtenir une classification non-supervisée des chirotopes de formes en trois dimensions (voir Partie V).

5.2.3 k -medoids

Dans cette section nous allons présenter la méthode de partitionnement des k -medoids. Cette méthode sera elle aussi utilisée pour classer les chirotopes lors de nos applications (voir Partie V).

Présentation de la méthode

Tout comme dans l'algorithme des k -means, le but de la méthode des k -medoids est de partitionner des données en un nombre k de clusters fixé à l'avance. La différence est sur le choix des points représentant les clusters : pour chaque cluster, dans l'algorithme des k -means on utilise un point moyen alors que dans la méthode des k -medoids on utilise un échantillon. Ainsi cette méthode est utilisable avec tout type de données. Les échantillons représentant les clusters sont appelés médoïdes et correspondent aux échantillons les plus "centraux" des classes. Dans cette méthode, on cherche les médoïdes permettant de minimiser le *critère de qualité* défini comme étant la somme pour chaque cluster des distances entre le médoïde du cluster et les points de ce cluster. Formellement, en notant Q le critère de qualité, on cherche à minimiser :

$$Q(E, \mathcal{C}_1, \dots, \mathcal{C}_k, m_1, \dots, m_k) = \sum_{j=1}^k \sum_{e_i \in \mathcal{C}_j} d(e_i, m_j)$$

On dira qu'un ensemble de k clusters $\mathcal{C} = \mathcal{C}_1, \dots, \mathcal{C}_k$ est engendré par k médoïdes m_1, \dots, m_k si les clusters sont créés de la façon suivante : Pour chaque échantillon e_i , déterminer le médoïde m_j le plus proche de e_i (ou en choisir un si le médoïde n'est pas unique) et affecter e_i au cluster \mathcal{C}_j . Formellement :

$$\mathcal{C}_j = \{e_i | d(e_i, m_j) = \min_k d(e_i, m_k)\} \quad \text{avec } e_i \text{ appartient exactement à un cluster.}$$

Il existe trois principaux algorithmes de clustering par la méthode des k -medoids appelés PAM, CLARA et CLARANS. Dans cette section, nous analysons ces algorithmes ainsi qu'un algorithme des k -medoids qui utilise la même méthode que dans les k -means pour la recherche de meilleurs médoïdes.

L'algorithme PAM (Partitioning Around Medoids) [36] est le suivant :

1. **Initialisation.** Choisir aléatoirement k échantillons m_1, \dots, m_k (appelés médoïdes et servant de représentants aux clusters).
2. **Création des clusters.** Générer les clusters $\mathcal{C}_1, \dots, \mathcal{C}_k$ engendrés par les médoïdes m_1, \dots, m_k .
3. **Calcul de la qualité.** On calcule la qualité $Q_1 = Q(E, \mathcal{C}_1, \dots, \mathcal{C}_k, m_1, \dots, m_k)$
4. **Recherche d'un meilleur médoïde.** Pour chaque médoïde m_j et pour chaque échantillon e_i qui n'est pas un médoïde :
 - Générer les clusters $\mathcal{C}'_1, \dots, \mathcal{C}'_k$ engendrés par $m_1, \dots, m_{j-1}, e_i, m_{j+1}, \dots, m_k$.
 - Calculer la qualité $Q_{i,j} = Q(E, \mathcal{C}'_1, \dots, \mathcal{C}'_k, m_1, \dots, m_{j-1}, e_i, m_{j+1}, \dots, m_k)$.

5. Soit $Q_{r,s} = \min_{i,j} Q_{i,j}$. Si $Q_{r,s} < Q_1$, remplacer Q_1 par $Q_{r,s}$ et m_s par e_r puis recommencer l'étape 4.

Tout comme pour l'algorithme des k -means, l'algorithme PAM converge vers un minimum local qui dépend de l'étape d'initialisation et la solution correspond au diagramme de Voronoï des médoïdes.

Nous allons étudier les temps de calculs de cet algorithme. L'algorithme PAM se base sur la matrice de distance entre tous les échantillons. Le calcul de cette matrice se fait en $O(n^2 \times l)$ où l est la longueur des vecteurs représentant les échantillons. Il est important de noter que la génération des clusters n'est pas indispensable à chaque itération : il suffit que dans le calcul de la qualité, on somme pour chaque échantillon la distance entre cet échantillon et le médoïde le plus proche. En procédant de cette façon, la seule chose qui demande des calculs à chaque itération est donc le calcul de la qualité pour chaque couple (m_j, e_i) où m_j est un médoïde et e_i est un échantillon qui n'est pas un médoïde. Dans la littérature, on trouve que pour chaque itération, l'algorithme réalise $O(k(n-k)^2)$ calculs. Par contre, n'ayant pas trouvé de preuve satisfaisante, nous avons prouvé cette complexité (dans [30] les auteurs justifient cette complexité en disant seulement que le calcul de la qualité se fait en $O(n-k)$). Nous allons montrer que l'algorithme effectue bien $O(k(n-k)^2)$ calculs par itération. Lorsque l'on remplacera le médoïde m_j par l'échantillon e_i , on notera $c(Q_2, m_j, e_i)$ le nombre d'opérations à effectuer pour calculer la qualité Q_2 et $c(m_j, e_i, e_h)$ le nombre d'opérations à effectuer pour calculer la distance minimale entre l'échantillon e_h et le médoïde le plus proche. Le nombre d'opérations par itération (noté $c(I)$) vaut donc

$$\begin{aligned} c(I) &= \sum_{i,j} c(Q_2, m_j, e_i) \\ &= \sum_{i,j,h} c(m_j, e_i, e_h). \end{aligned}$$

Le nombre d'opérations $c(m_j, e_i, e_h)$ dépend du médoïde le plus proche de l'échantillon e_h avant de remplacer m_j par e_i . En effet, si le médoïde le plus proche de e_h était m_r avec $r \neq j$ avant de remplacer m_j par e_i alors après ce remplacement, le médoïde le plus proche de e_h est soit m_r soit e_i . Par contre, si le médoïde le plus proche de e_h était m_j avant de remplacer m_j par e_i alors après ce remplacement le médoïde le plus proche de e_h peut être e_i ou n'importe quel médoïde m_r avec $r \neq j$. En notant m_s le médoïde le plus proche de e_l avant de remplacer m_j par e_i et $d_{min}(e_h)$ la distance minimale entre l'échantillon e_h et le médoïde le plus proche après le remplacement, on a donc :

$$d_{min}(e_l) = \begin{cases} \min(d(e_i, e_l), d(m_s, e_l)) & \text{si } s \neq j \\ \min(d(e_i, e_l), \min_{r \neq j} d(m_r, e_l)) & \text{si } s = j \end{cases}$$

Ainsi le nombre d'opérations $c(m_j, e_i, e_l)$ est de $O(k)$ si m_j était le médoïde le plus proche de e_l avant de remplacer m_j par e_i et le nombre d'opérations est de $O(1)$ sinon. Le nombre d'opérations

par itération $c(I)$ vaut alors

$$\begin{aligned}
 c(I) &= \sum_{i,j,l} c(m_j, e_i, e_l) \\
 &= \sum_{i,l} \sum_j c(m_j, e_i, e_l) \\
 &= O((n-k)^2 \times (1 \times k + (k-1) \times 1)) \\
 &= O((n-k)^2 \times k).
 \end{aligned}$$

En parallèle, les auteurs de l'algorithme PAM ont créé l'algorithme CLARA pour traiter de gros jeux de données [37]. Cet algorithme utilise un sous-ensemble aléatoire des échantillons pour déterminer les médoïdes, puis classe l'ensemble des échantillons à partir de ces médoïdes. L'algorithme calcule ensuite le critère de qualité pour ces clusters et recommence pour un autre sous-ensemble aléatoire des échantillons. Après avoir recommencé un certain nombre de fois (fixé par l'utilisateur), l'algorithme retourne les médoïdes et les clusters associés qui ont le critère de qualité le plus faible.

De nombreux autres algorithmes se basent sur les médoïdes pour obtenir un clustering d'un ensemble de données [50, 54, 80, 88]. Parmi ces algorithmes, l'algorithme CLARANS [50] est très semblable à l'algorithme PAM, mais au lieu de chercher un meilleur médoïde parmi toutes les paires possibles (médoïde, non-médoïde), la recherche n'a lieu que pour un nombre fixé à l'avance de paires. De plus, pour limiter le temps d'exécution, l'algorithme CLARANS ne réalise qu'un nombre fixé d'itérations. D'après les auteurs, cet algorithme est aussi efficace que PAM pour les petits jeux de données tout en étant plus rapide et est plus efficace que CLARA pour les grands jeux de données. La limitation de cet algorithme est le fait qu'il y ait deux paramètres à fixer a priori : le nombre de paires à traiter à chaque itération et du nombre d'itérations.

Récemment un algorithme a été proposé pour trouver des clusters à l'aide de médoïdes. Cet algorithme utilise le même principe que dans les k -means pour la recherche de meilleurs médoïdes [54]. Il s'agit de trouver pour chaque cluster \mathcal{C}_j l'échantillon qui minimise la somme des distances aux autres échantillons de \mathcal{C}_j . Ainsi on ne teste pas de remplacer un médoïde par l'un des $(n-k)$ échantillon non-médoïde mais uniquement par un des échantillons du même cluster, ce qui diminue le nombre d'opérations. Une applet [47] permet de comparer cette méthode à la méthode des k -means. Grâce à cet applet, nous avons trouvé un exemple en deux dimensions représenté par la Figure 5.6 tel que, si les clusters ne sont pas bien initialisés et qu'il y a des zones du plan sans point alors cet algorithme ne permet pas d'avoir les bons clusters contrairement aux k -means. En effet dans une telle situation, lors d'itérations de l'algorithme des k -means le point moyen sera dans une zone du plan sans point. Or avec cet algorithme des k -medoids, le médoïde d'une classe restera toujours du même côté de la zone sans point, ce qui empêche d'avoir la solution optimale.

Avantages et inconvénients pour la classification de chirotopes

La méthode des k -medoids peut être utilisée pour n'importe quel type de données et permet d'utiliser n'importe quelle distance. Ainsi cette méthode peut être utilisée pour classer des chirotopes de formes en deux ou trois dimensions sans que l'on ait besoin de transformer les données.

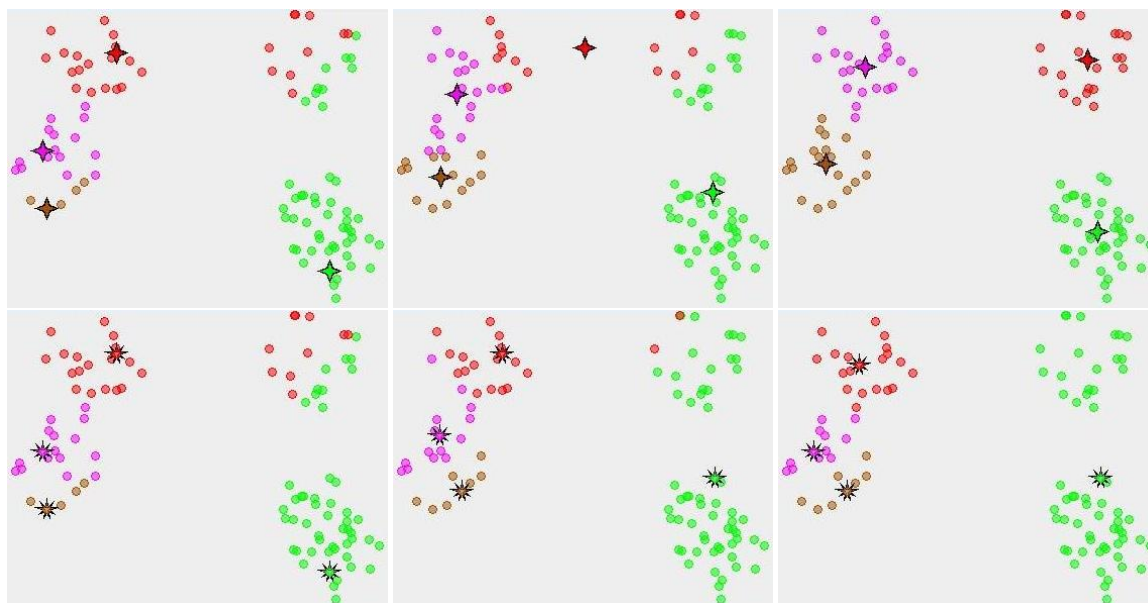


FIGURE 5.6 – Comparaison entre l’algorithme des k -means (première ligne) et l’algorithme basé sur les k -medoids qui s’en rapproche [54] (deuxième ligne) pour classer en quatre clusters (vert, rouge, violet et marron) un même ensemble de points en deux dimensions. Les deux algorithmes partent de la même initialisation dans le but d’obtenir quatre clusters. Les croix sur la première ligne représentent les points moyens et les étoiles de la deuxième ligne représentent les médoides. La colonne de gauche représente l’étape d’initialisation, celle du milieu représente l’état au milieu du processus et celle de droite représente le résultat fourni par les algorithmes. Les images ont été obtenues à l’aide de l’applet [47].

Cette méthode dispose en plus de deux avantages. Le premier avantage est que contrairement aux k -means cette méthode est combinatoire. En effet, pour chaque cluster il n’y a qu’un nombre fini de représentant possible puisque le représentant de chaque cluster est l’un des chirotopes (contrairement à la méthode des k -means qui utilise le point moyen).

Le deuxième avantage est que la complexité de l’algorithme PAM ne dépend pas de la taille des chirotopes. Ainsi une fois que la matrice des distances est calculée, la méthode des k -medoids ne dépend plus que du nombre n de chirotopes et du nombre k de clusters. Or dans nos applications le nombre de formes en trois dimensions à comparer est assez petit (de l’ordre de 300 pour les applications comptant le plus de formes) alors que le nombre de bases est bien plus important (près de 2 700 000 bases non-fixes pour les applications comptant le plus de bases). Ainsi la méthode des k -medoids sera rapide à calculer.

Par contre cette méthode présente deux inconvénients. Le premier inconvénient est que cette méthode peut dans certains cas ne pas donner la bonne classification, alors que la méthode des k -means aurait donné la bonne classification comme le montre la Figure 5.7.

Le deuxième inconvénient de cette méthode est que l’on ne peut pas obtenir une caractérisation des clusters qui se fonde sur des $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$. En effet, la seule caractérisation d’un

cluster \mathcal{C}_i que l'on peut avoir est qu'un échantillon de \mathcal{C}_i est plus proche du médoïde de \mathcal{C}_i que de n'importe quel médoïde d'une autre classe. Cette caractérisation n'utilise qu'une distance globale et ne permet donc pas de trouver un sous-ensemble de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ caractérisant chaque cluster.



FIGURE 5.7 – Exemple de situations où le résultat des k -medoids (à gauche) est moins bon que le résultat des k -means (à droite). Les croix sur l'image de droite représentent les points moyens et les étoiles de l'image de gauche représentent les médoïdes.

Lorsque nous étudions un ensemble d'échantillons, nous commençons par calculer la matrice des distances entre tous les chirotopes. Or nous avons vu que le temps de calcul pour obtenir une classification par la méthode des k -medoids est faible (par rapport aux k -means) si nous possédons déjà la matrice des distances. Comme la classification obtenue est parfois moins bonne que celle obtenue par la méthode des k -means, nous utiliserons généralement l'algorithme de PAM pour déterminer rapidement si notre méthode basée sur les matroïdes orientés peut s'appliquer à un jeu de données et s'il est donc utile de calculer une classification à l'aide de l'algorithme des k -means (voir Partie V).

5.2.4 Analyse d'autres méthodes de clustering

Il existe de nombreuses autres méthodes de partitionnement. La méthode se rapprochant le plus des k -means et k -medoids est la méthode des k -medians [34]. Il existe parfois des confusions entre cette méthode et celle des k -medoids. Le principe est de partitionner les données en k clusters en choisissant comme représentant pour chaque cluster le point de \mathbb{R}^l (avec l la taille des vecteurs) où chaque coordonnée est la médiane des coordonnées des échantillons du cluster.

Nous avons choisi de ne pas utiliser cette méthode. En effet, lorsque l'on code les chirotopes par des points, chaque coordonnée de nos points ne peut prendre que deux valeurs (-1 et 1 dans le cas uniforme et 0 et 1 dans le cas non-uniforme). Ainsi dans la méthode des k -médians, les représentants des clusters correspondent aux points moyens que l'on aurait arrondis. Cette méthode ne permet pas de bénéficier de ce qui est pour nous l'avantage principal des k -means, c'est-à-dire de pouvoir calculer simplement à partir des représentants des clusters, la proportion de chirotopes ayant un signe donné pour chaque $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ dans ce cluster. De plus, bien qu'étant combinatoire, cette méthode ne possède pas le deuxième avantage de la méthode des k -medoids puisque son temps de calcul est le même que celui de la méthode des k -means.

Toutes les méthodes précédentes construisent les clusters autour de points (un par cluster). D'autres méthodes existent tels que les méthodes basées sur la densité [12, 16, 38, 70], qui per-

mettent de trouver des clusters ayant des formes quelconques comme par exemple des clusters formant des cercles concentriques dans le plan. Le plus souvent ces algorithmes ne nécessitent pas un nombre de classes en paramètre.

Parmi ces méthodes, l'un des algorithmes les plus connus est l'algorithme DBSCAN [16]. Cet algorithme utilise deux paramètres : le rayon r et le nombre de points minimum $pMin$. Le principe de l'algorithme est le suivant : on prend un point p non visité et on compte le nombre de points dans la boule centrée en p et de rayon r . Si ce nombre de points est supérieur à $pMin$ alors on crée un nouveau cluster contenant p et tous les points dans la boule. Ensuite on crée une liste L de points à regarder contenant tous les points de la boule (sauf p) qui nous servira à voir si l'on peut étendre ce cluster. Pour tous les points de la liste, si la boule de rayon r centrée en ce point contient au moins $pMin$ points, on ajoute ces points à la liste L et au cluster. Si la boule contient moins de $pMin$ points alors on ajoute juste ces points au cluster. Quand on a visité tous les points de la liste, on recommence avec un point non visité. Les résultats de cet algorithme sont sensibles aux choix des paramètres [49]. Par contre cet algorithme ne classe pas les points aberrants.

Dans les méthodes basées sur la densité, il existe également la méthode Mean Shift [12]. Cette méthode est une méthode itérative qui utilise comme seul paramètre un rayon r . Le principe est le suivant : on part d'un point p et on calcule le point moyen de l'ensemble des points situés dans la boule de rayon r centré en p . On déplace alors p vers le point moyen et on recommence (on calcule le point moyen et on déplace p) jusqu'à ce que le point p ne bouge plus. On replace alors p à sa position initiale et on recommence avec un autre point. Un cluster sera alors formé de tous les points qui ont été déplacés vers un même point d'arrivée.

Nous n'avons pas utilisé ces méthodes puisque nos jeux de données sont de petites tailles (de 60 à 300 échantillons) ce qui fait que les clusters ont de fortes chances de ne pas être denses. De plus ces méthodes ne nous permettent pas d'obtenir une caractérisation des classes.

Des méthodes spécifiques à un type de donnée existent également. Par exemple, un algorithme de clustering utilise des calculs matriciels [42] pour regrouper les échantillons et les variables en clusters lorsque les données sont des données binaires. L'idée générale de cet algorithme est de réaliser des permutations sur des lignes et des colonnes de la matrice des données (dans laquelle chaque ligne représente un échantillon). Le but de ces permutations est d'obtenir à la fin que la matrice soit sous forme d'une matrice diagonale par blocs, c'est-à-dire que l'on peut diviser cette matrice en sous-matrices de sorte qu'une sous-matrice soit composée majoritairement de 1 si elle est sur la diagonale de la matrice et qu'elle soit composée majoritairement de 0 si elle n'est pas sur la diagonale de la matrice.

Nous n'avons pas utilisé cette méthode car elle consiste à regrouper les échantillons qui ont beaucoup de 1 aux mêmes coordonnées. Pour cela, l'algorithme utilise la distance de Jaccard que nous avons décrite dans la Section 5.1 et nous avons alors vu que cette distance ne pouvait pas être utilisée pour comparer des chirotopes.

5.3 Méthodes de classification supervisée

Dans les méthodes de clustering, n'ayant aucune information sur la composition des clusters, toutes les variables sont considérées de la même façon. Par contre dans la classification supervisée

une partie des données est déjà classées. Les méthodes de classification supervisée commencent donc par une phase d'apprentissage qui consiste à créer un modèle de classification à partir des données déjà classées. Ce modèle de classification doit non seulement classer les données utilisées lors de la phase d'apprentissage mais doit également être capable de prédire correctement la classe des autres échantillons. On cherche donc à minimiser le nombre d'erreurs lors de la classification. Selon le domaine d'application il peut être nécessaire de ne pas considérer les différentes erreurs de la même façon. C'est le cas par exemple dans le domaine médical où l'erreur faite en classant un patient sain comme étant malade n'est pas équivalente à l'erreur faite en classant un patient malade comme étant sain. Dans ce cas il faut alors ajouter une pondération lors de l'estimation des erreurs effectuées.

Dans cette section, nous nous concentrons sur une méthode de classification supervisée couramment utilisée : les arbres de décision, puis nous analysons deux autres méthodes que sont la classification naïve bayésienne et les réseaux de neurones artificiels. Nous présentons chacune de ces méthodes et expliquons au fur et à mesure pourquoi nous avons choisi de ne pas utiliser ces méthodes.

5.3.1 Arbres de décision

Présentation de la méthode

Un arbre de décision est une façon de représenter une classification à l'aide d'un arbre. Le principe de cette méthode est de séparer les données en sous-ensembles d'échantillons de façon récursive afin que chaque sous-ensemble ne soit constitué presque uniquement d'échantillons d'une même classe. A chaque étape la séparation se fait en posant des conditions sur les valeurs d'une seule variable. Lors du processus d'apprentissage, l'algorithme détermine donc les variables qui permettent de séparer au mieux les classes prédéfinies. Ces variables correspondront aux nœuds internes de l'arbre de décision. Si un nœud interne v_i correspond à une variable y_j alors les arêtes entre le nœud v_i et ses fils correspondent aux différentes valeurs prises par la variable y_j . Ainsi une arête peut correspondre à une valeur ou un sous ensemble de valeurs (variables discrètes) ou à un intervalle (variables continues). Les feuilles de l'arbre correspondent aux classes : une feuille correspond à une seule classe mais plusieurs feuilles peuvent correspondre à une même classe. Ainsi dans l'arbre de décision de la Figure 5.8, deux feuilles correspondent à la classe \mathcal{C}_1 et deux feuilles correspondent à la classe \mathcal{C}_2 .

Un arbre de décision peut s'utiliser de deux façons. La première utilisation nous permet de classer un nouvel échantillon e . On part alors de la racine de l'arbre qui correspond à une variable (que l'on note y_r). On parcourt l'arête correspondant à la valeur $e(r)$ pour arriver sur un nouveau nœud. Tant que ce nœud n'est pas une feuille, on continue de la même façon. Lorsque l'on arrive à une feuille, elle nous indique comment classer l'échantillon e .

La deuxième utilisation consiste à trouver une discrimination des différentes classes. Lorsque l'on cherche à caractériser une classe, on remonte l'arbre depuis les feuilles correspondant à cette classe vers la racine de l'arbre. Par exemple l'arbre de décision de la Figure 5.8 caractérise la classe \mathcal{C}_1 de la façon suivante :

$$e_i \in \mathcal{C}_1 \iff (e_i(1) \geq \alpha_1 \text{ ET } e_i(1) < \alpha_3) \quad \text{OU} \quad (e_i(1) < \alpha_1 \text{ ET } e_i(1) \geq \alpha_2 \text{ ET } e_i(1) \geq \alpha_3).$$

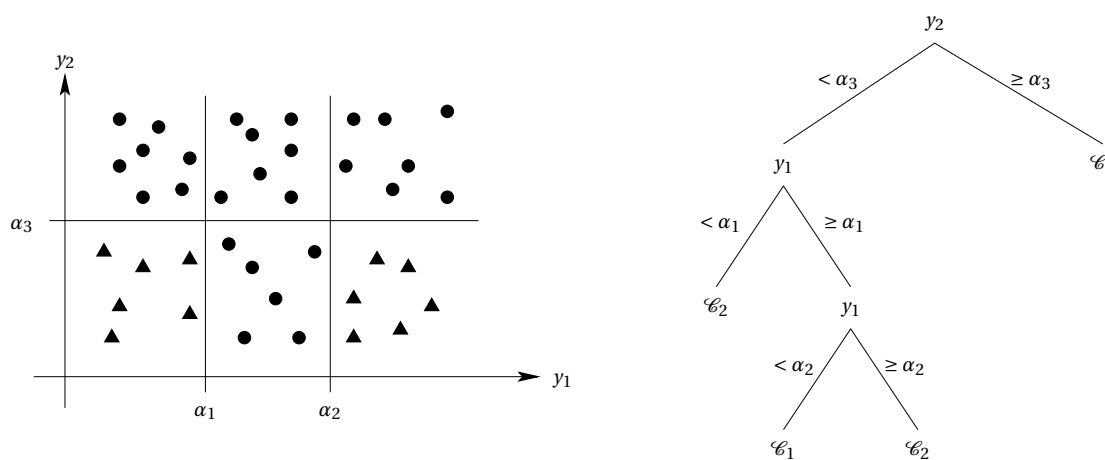


FIGURE 5.8 – Exemple d’un arbre de décision (à droite) créé à partir de points en 2 dimensions (à gauche) répartis en deux classes. La classe \mathcal{C}_1 correspond aux ronds et la classe \mathcal{C}_2 correspond aux triangles.

Un “bon” arbre de décision doit donc satisfaire deux critères : permettre de classer les échantillons avec un minimum d’erreur et proposer une caractérisation des classes qui ne soit pas trop compliquée. Le problème est donc de déterminer la hauteur de l’arbre car plus un arbre sera petit et plus la caractérisation sera simple mais moins la classification risque d’être précise.

La création d’un arbre de décision nécessite l’utilisation de trois opérateurs. Le premier opérateur permet de décider si un nœud correspond ou non à une feuille, c’est-à-dire si le sous-ensemble d’échantillons correspondant à ce nœud doit encore être divisé ou si la quasi-totalité de ces échantillons font partie d’une même classe. Si le nœud est un nœud interne de l’arbre, le deuxième opérateur permet de déterminer quelle variable doit être utilisée pour séparer le sous-ensemble d’échantillon correspondant à ce nœud. Si le nœud est une feuille, le troisième opérateur est utilisé pour affecter une classe à une feuille.

Pour le premier opérateur, il existe plusieurs fonctions permettant de calculer si un sous-ensemble d’échantillons correspondant à un nœud doit encore être divisé ou non. En notant S le sous-ensemble d’échantillon associé à un nœud et n_i le nombre d’échantillons de S appartenant à la classe \mathcal{C}_i , les deux fonctions les plus utilisées pour calculer cela sont les suivantes :

- Coefficient de Gini : $G(S) = 1 - \sum_{1 \leq i \leq k} (n_i/|S|)^2$
- Indice d’entropie de Shannon : $H(S) = - \sum_{1 \leq i \leq k} (n_i/|S|) \times \ln(n_i/|S|)$

Plus la valeur de ces fonctions est grande et plus les échantillons de S sont répartis entre plusieurs classes et ces fonctions atteignent leur minimum lorsque tous les échantillons de S appartiennent à une seule classe. Ainsi, avant de construire l’arbre il faut faire un choix sur la fonction qui sera utilisée pour le premier opérateur.

Ensuite, sachant qu’il n’existe peut-être pas d’arbres de décision permettant de classer les données sans erreurs, il faut alors déterminer un seuil σ_0 sur la valeur de la fonction. Si la valeur est supérieure au seuil on devra séparer le sous-ensemble d’échantillons et le nœud sera alors un nœud

interne ; sinon le nœud sera une feuille. Le seuil déterminera donc la hauteur de l'arbre de décision. Généralement, pour limiter la hauteur de l'arbre, on décide également qu'un nœud sera une feuille si la taille de l'ensemble d'échantillons S est plus petit qu'une certaine taille t_0 . Ainsi le premier opérateur nécessite deux paramètres, le seuil σ_0 et la taille t_0 . Ces deux paramètres jouent un rôle important sur la taille de l'arbre et donc sur le résultat.

Le troisième opérateur est le plus simple. On affecte à une feuille la classe qui est majoritaire parmi tous les échantillons du sous-ensemble d'échantillons correspondant à la feuille. Par contre le deuxième opérateur dépend de l'algorithme utilisé pour construire l'arbre de décision.

Il existe de nombreux algorithmes pour construire un arbre de décision. Parmi ces algorithmes, on peut citer l'algorithme CART [9] (pour "Classification And Regression Trees") qui utilise le coefficient de Gini pour créer un arbre de décision binaire (c'est-à-dire dont tous les nœuds internes ont exactement deux fils). Dans cet algorithme, le deuxième opérateur servant à déterminer quelle variable va être utilisée pour séparer un nœud interne repose sur la fonction $\Delta(S, v)$ définie comme ceci : notons S le sous-ensemble d'échantillons correspondant au nœud interne et $S_1(v)$ et $S_2(v)$ les deux sous-ensembles de S formés en utilisant la variable v pour séparer S , on a alors $\Delta(S, v) = G(S) - (G(S_1(v)) \times |S_1|/|S|) + G(S_2(v)) \times |S_2|/|S|$. Le deuxième opérateur consiste alors à choisir la variable qui maximise $\Delta(S, v)$.

Deux autres algorithmes très souvent utilisés sont l'algorithme ID3 [64] et son amélioration C4.5 [65], tous les deux proposés par Quinlan. Les principales améliorations apportées dans C4.5 sont le traitement des variables continues en créant une valeur de séparation, la gestion des données manquantes (ces valeurs ne sont alors pas prises en compte) et l'élagage de l'arbre (nous expliquerons ce qu'est l'élagage dans la suite).

Nous allons voir comment est défini le deuxième opérateur dans l'algorithme C4.5. Pour déterminer quelle variable va être utilisée pour séparer un ensemble d'échantillon S correspondant à un nœud interne en r sous-ensembles de S , on calcule la fonction $\Delta(S, v, r)$ définie comme ceci : notons $S_1(v), \dots, S_r(v)$ les r sous-ensembles de S formés en utilisant la variable v pour séparer S , on a alors $\Delta(S, v, r) = Gain(S, v, r) / (\sum_{1 \leq i \leq r} (|S_i(v)|/|S|) \times \ln(|S_i(v)|/|S|))$ où $Gain(S, v, r) = H(S) - \sum_{1 \leq i \leq r} G(S_i(v))$. Le deuxième opérateur consiste alors à choisir la variable qui maximise $\Delta(S, v, r)$. Dans l'algorithme ID3, le deuxième opérateur consiste à choisir la variable qui maximise $Gain(S, v, r)$. Or plus r est grand, plus la valeur de la fonction $Gain(S, v, r)$ a tendance à être grande. La division dans le calcul de $\Delta(S, v, r)$ corrige ce problème.

Quelle que soit la méthode employée, après la construction de l'arbre de décision il y a une phase d'élagage de l'arbre. Le principe de l'élagage est de diminuer la hauteur de l'arbre en supprimant des sous-arbres issus de nœuds internes ; ces nœuds deviennent alors des feuilles. Cette étape est nécessaire car les feuilles qui sont créées ont tendance à correspondre à des sous-ensembles d'échantillons de petites tailles. De plus, sans élagage, les arbres sont souvent trop spécialisés sur les données d'apprentissage. La Figure 5.9 tirée de [76] montre ce phénomène. En effet, on remarque sur ce graphique que le nombre d'erreurs de classification sur les données d'apprentissage diminue avec le nombre de feuilles alors que ce nombre d'erreurs sur des données de test augmente à partir d'un certain nombre de feuilles. Ainsi, si l'on essaye de classer un échantillon n'apparaissant pas dans les données d'apprentissage, il arrive que ce nouvel échantillon soit mal classé.

Pour procéder à l'élagage, les méthodes les plus utilisées reposent sur une estimation du taux d'erreurs de classement si on utilisait l'arbre de décision créé pour classer un nouveau jeu de don-

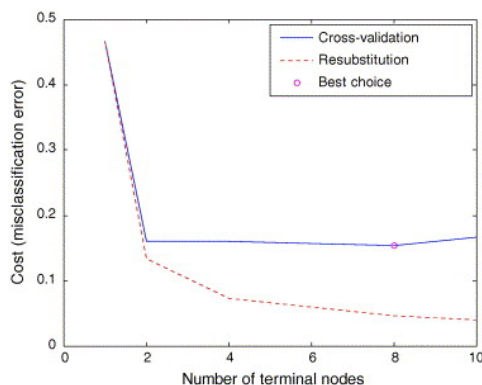


FIGURE 5.9 – Graphique tiré de [76] représentant le nombre d’erreurs de classification sur les données d’apprentissage (en rouge) et sur les données de test (en bleu) en fonction du nombre de feuilles.

nées. Pour réaliser cette estimation, l’algorithme CART calcule le taux d’erreur sur un jeu de données déjà classées et qui n’a pas été utilisé pour l’apprentissage. Par contre l’algorithme C4.5 utilise les données d’apprentissage pour estimer le taux d’erreur.

Cette diminution de la hauteur de l’arbre est une étape très importante car en plus de diminuer les erreurs de classement, elle permet d’accélérer l’utilisation de l’arbre pour faire de la classification et permet également de simplifier la caractérisation des classes.

Avantages et inconvénients pour la classification de chirotopes

La classification supervisée par des arbres de décision est une méthode permettant à la fois de classer et de discriminer les classes. De plus cette méthode à l’avantage de fournir des résultats facilement compréhensibles. Comme les arbres de décision peuvent être utilisés sur tous les types de données et sont particulièrement bien adaptés aux données discrètes, ils peuvent être appliqués à la classification de chirotopes tout en fournissant une discrimination des classes.

Les arbres de décision présentent en plus deux avantages. Le premier avantage est de ne pas être perturbé par des échantillons atypiques. Ces derniers seront peut-être mal classés mais ils ne perturberont pas la classification des autres échantillons contrairement à la méthode des k -means par exemple (c.f. Section 5.2) dont les coordonnées des échantillons atypiques ont une influence directe dans la position du point moyen des classes.

Le deuxième avantage des arbres de décision est que la classification par les arbres de décision ne fait aucune hypothèse sur la répartition des échantillons. En effet, dans les méthodes des k -means et k -medoids on suppose que la répartition des échantillons correspond à un diagramme de Voronoï. Dans d’autres méthodes, on suppose que les échantillons sont répartis suivant une loi de probabilité. De telles méthodes ne sont donc pas adaptées pour la classification d’échantillons si ceux-ci ne respectent pas de telles hypothèses de répartition. Ainsi les arbres de décision peuvent être utilisés si l’on ne connaît pas a priori la répartition des échantillons.

Par contre la classification par les arbres de décision présente deux inconvénients importants. Nous venons de voir que dans tous les algorithmes, pour un nœud donné, la variable qui va être

utilisée pour séparer ce nœud maximise une certaine fonction. Or ce maximum peut être atteint pour plusieurs variables. Dans ce cas, les algorithmes ne peuvent pas savoir quelle(s) variable(s) donnera le meilleur arbre de décision. Il faudrait générer tous les arbres de décision possible en utilisant pour chaque nœud toutes les variables maximisant la fonction. Cela risque d'être coûteux en temps de calcul s'il y a plusieurs nœuds pouvant chacun utiliser plusieurs variables, ce qui nécessiterait ensuite un tri pour ne conserver que les arbres minimisant les erreurs de classification.

Le deuxième inconvénient des arbres de décision est que le choix d'une variable pour décomposer un nœud ne changera jamais et ne sera jamais remise en question. En effet, lorsque un algorithme décide d'utiliser une variable pour séparer un nœud v en r sous-ensembles, l'algorithme va créer r fils au nœud v puis va travailler avec un nœud qui n'est pas un nœud interne. Le seul moment où l'algorithme traitera de nouveau le nœud v sera au moment de l'élagage pour déterminer si v reste un nœud interne ou si l'on supprime les fils de v (et donc que v redevienne une feuille). A aucun moment l'algorithme essaiera de décomposer le nœud v en d'autres sous-ensembles. Or il arrive que la meilleure décision pour la décomposition consiste à utiliser successivement deux variables sans qu'aucune ne maximise la fonction. Ainsi quel que soit l'algorithme, l'arbre de décision obtenu ne sera pas le meilleur arbre possible et n'indiquera pas que ces deux variables prises successivement permettent de bien classer les échantillons.

Les codes de l'algorithme C4.5 étant disponible en ligne et les arbres de décision étant implémentés dans de nombreux logiciels (comme par exemple dans de nombreuses bibliothèques du logiciel R <http://www.r-project.org/> ou dans le logiciel Weka <http://www.cs.waikato.ac.nz/ml/weka/index.html>), nous avons essayé d'utiliser les arbres de décision sur le jeu de données que nous présentons au Chapitre 11, pour déterminer si les classes fournies par des experts pouvaient être discriminées facilement. Nous avons analysé ce jeu de données en deux temps, d'abord en utilisant tous les landmarks (voir Section 11.1) puis en n'utilisant que les landmarks anatomiques (voir Section 11.2).

Dans la première étude, nous nous sommes aperçu qu'un grand nombre de variables (bases) maximisaient la fonction pour tous les nœuds de l'arbre. Par exemple sur les 2 675 999 bases non-fixes que compose chaque chirotopes, nous avons estimé qu'il y avait de l'ordre de 21 000 bases qui pouvaient être utilisées pour le nœud racine de l'arbre. Comme nous voulons obtenir toutes les façons de caractériser les classes des experts, il nous fallait générer tous les arbres possibles. Nous avons alors décidé de ne pas utiliser les arbres de décision car le nombre d'arbres à étudier était trop important et aurait nécessité de trop grands temps de calcul.

Dans la deuxième étude, l'algorithme C4.5 ne choisissait pas la bonne variable pour un nœud car la variable permettant la meilleure classification ne maximisait pas la fonction. En effet, la Figure 11.9 indique qu'il faut utiliser deux bases pour donner une caractérisation de la classe BCS. Or aucune de ces deux bases ne maximisait pas la fonction. L'algorithme utilisait à la place une autre base. L'arbre de décision retourné par l'algorithme ne classait pas bien certains échantillons de la classe BCS.

Au vu des deux inconvénients précédents nous avons décidé de ne pas utiliser les arbres de décision que ce soit pour faire de la classification supervisée ou pour trouver plusieurs façons de discriminer les classes. Par contre nous nous sommes inspirés de cette méthode (et notamment de l'idée de trier les variables selon une certaine fonction) pour créer notre méthode présentée dans le Chapitre 6.

5.3.2 Classification naïve bayésienne

Présentation de la méthode

La classification naïve bayésienne est une méthode de classification supervisée basée sur des calculs de probabilité en considérant que les variables sont indépendantes (d'où le terme naïf). Cette méthode de classification tire son nom du théorème de Bayes qui traite de la probabilité conditionnelle.

Pour expliquer la méthode nous allons poser quelques notations :

- l : le nombre de variables de chaque échantillon
- $P(e \in \mathcal{C}_j | e(x_1), \dots, e(x_l))$: la probabilité qu'un échantillon e appartienne à \mathcal{C}_j en prenant en compte les valeurs de e pour les variables $\{x_1, \dots, x_l\}$
- $P(e \in \mathcal{C}_j)$: la probabilité a priori que e appartienne à \mathcal{C}_j (a priori dans le sens où l'on ne connaît rien de e et donc on n'utilise pas les valeurs de e pour les différentes variables)
- $P(e(x_1), \dots, e(x_l) | e \in \mathcal{C}_j)$: la probabilité que les valeurs d'un échantillon e pour les variables x_1, \dots, x_l soient $e(x_1), \dots, e(x_l)$ sachant que $e \in \mathcal{C}_j$
- $P(e(x_1), \dots, e(x_l))$: la probabilité a priori que les valeurs d'un échantillon e pour les variables x_1, \dots, x_l soient $e(x_1), \dots, e(x_l)$ (a priori dans le sens où l'on ne sait pas dans quelle classe est l'échantillon e)

Si l'on essaye de classer un nouvel échantillon e après l'apprentissage sur un jeu de données (donc e ne fait pas partie du jeu de données d'apprentissage), alors cette méthode consiste à calculer pour chaque classe \mathcal{C}_j la probabilité $P(e \in \mathcal{C}_j | e(x_1), \dots, e(x_l))$ et à classer e dans la classe maximisant cette probabilité. En appliquant le théorème de Bayes à notre problème de classification on obtient alors :

$$P(e \in \mathcal{C}_j | e(x_1), \dots, e(x_l)) = \frac{P(e \in \mathcal{C}_j) \times P(e(x_1), \dots, e(x_l) | e \in \mathcal{C}_j)}{P(e(x_1), \dots, e(x_l))}$$

Comme nous avons supposé que les variables sont indépendantes, on obtient :

$$P(e \in \mathcal{C}_j | e(x_1), \dots, e(x_l)) = \frac{P(e \in \mathcal{C}_j) \times \prod_{1 \leq i \leq l} P(e(x_i) | e \in \mathcal{C}_j)}{P(e(x_1), \dots, e(x_l))}$$

Le dénominateur ne dépend pas de la classe \mathcal{C}_j , il est donc le même quelle que soit la classe. Ainsi on se contente de calculer le numérateur pour chaque classe.

Pour calculer la probabilité a priori que e appartienne à la classe \mathcal{C}_j ($P(e \in \mathcal{C}_j)$), il faut poser des hypothèses sur la distribution des échantillons dans les différentes classes. On peut par exemple supposer que nous sommes dans un cas d'équiprobabilité (c'est-à-dire que la distribution est la même pour chaque classe), ou que la distribution dans le cas général est la même que pour les données d'apprentissage.

Pour le calcul de $P(e(x_i) | e \in \mathcal{C}_j)$, si les variables prennent des valeurs continues on utilise une densité de probabilité en faisant donc une hypothèse sur la loi de probabilité que suivent ces valeurs. Dans le cas discret on peut calculer directement cette probabilité. Elle est égale au nombre d'échantillons de \mathcal{C}_j qui ont la même valeur que e pour la variable x_i divisée par le nombre d'échantillons dans \mathcal{C}_j . Comme $P(e(x_i) | e \in \mathcal{C}_j)$ peut être nulle, on ajoute une constante appartenant à $]0, 1[$ à $P(e(x_i) | e \in \mathcal{C}_j)$ pour éviter que le numérateur (et donc $P(e \in \mathcal{C}_j | e(x_1), \dots, e(x_l))$) soit nul.

Avantages et inconvénients pour la classification de chirotopes

Il y a deux raisons qui ont fait que nous n'avons pas utilisé cette méthode. La première est que plusieurs études ont montré que les arbres de décision sont plus performants que la méthode de classification naïve bayésienne [46, 81] lorsque les données sont binaires. Or nous avons vu dans la Section 5.1 qu'on pouvait transformer simplement des chirotopes en données binaires. La deuxième raison est que cette méthode ne nous permet pas de caractériser les classes par un sous-ensemble de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$. En effet, pour déterminer dans quelle classe se trouve un échantillon, on réalise une multiplication des probabilités pour toutes les bases, ce qui ne nous permet pas de remonter à un sous-ensemble de bases.

5.3.3 Réseaux de neurones artificiels

Présentation de la méthode

La méthode de classification supervisée basée sur les réseaux de neurones artificiels a pour but de simuler le fonctionnement du cerveau humain. Le cerveau humain est constitué de neurones qui sont interconnectés en réseau. Lorsque le cerveau reçoit une information de l'un des cinq sens, cette information arrive sous forme d'impulsions électriques. Lorsqu'un neurone reçoit suffisamment d'impulsions électriques, ce neurone envoie des impulsions aux neurones qui lui sont connectés.

Pour modéliser ce principe, chaque impulsion électrique est remplacée par une valeur numérique. Un neurone est activé si la somme des valeurs qu'il reçoit est supérieure à une certaine valeur de seuil. Si le neurone est activé, il transmet la somme des valeurs qu'il a reçues aux autres neurones. Les réseaux de neurones artificiels sont composés de neurones positionnés en différentes couches. Généralement, la première couche contient autant de neurones qu'il y a de coordonnées dans nos échantillons et la dernière couche contient autant de neurones qu'il y a de classes. Le j -ème neurone de la première couche reçoit la valeur de la j -ème coordonnée de l'échantillon fourni en entrée. S'il est activé, il envoie cette valeur à tous les échantillons de la deuxième couche. Chaque connexion entre deux neurones est pondérée par un poids strictement positif. Lorsqu'une valeur doit transiter par cette connexion, elle est alors multipliée par ce poids. Les informations de l'échantillon en entrée transitent alors de couche en couche jusqu'à la dernière couche. Si le réseau est bien adapté, la dernière couche ne doit contenir qu'un seul neurone actif qui indique alors à quelle classe correspond cet échantillon.

Lors de la phase d'apprentissage, les poids sont initialisés de façon aléatoire. On présente ensuite en entrée du réseau un échantillon d'apprentissage. Si la sortie du réseau ne correspond pas à la classe de cet échantillon, les poids des connexions entre les neurones sont modifiés.

Avantages et inconvénients pour la classification de chirotopes

Les réseaux neuronaux artificiels sont parmi les méthodes les plus performantes pour classer des jeux de données [46]. Par contre ils présentent notamment deux inconvénients. Le premier est qu'il est très difficile de choisir le nombre de couches (généralement une ou deux couches internes suffisent) ainsi que le nombre de neurones nécessaires [46, 77]. De plus, cette méthode utilise de

nombreux autres paramètres (comme par exemple le seuil de chaque neurone) qu'il faut déterminer pour pouvoir utiliser cette méthode et les résultats sont très sensibles à tous ces paramètres. Il est très difficile d'obtenir les bonnes valeurs pour ces paramètres car la valeur d'un paramètre pour obtenir le meilleur résultat possible n'est pas la même selon que l'on fasse varier que ce paramètre ou tous les paramètres. Ainsi l'utilisation des réseaux neuronaux demande une certaine expérience afin d'obtenir des résultats satisfaisants.

Le deuxième inconvénient de cette méthode est qu'il n'est pas possible d'obtenir une caractérisation simple des classes. En effet, il ne paraît pas possible de connaître l'influence d'une variable dans le processus de classification quand le réseau de neurone compte au moins une couche interne.

Ces inconvénients nous ont conduits à ne pas utiliser les réseaux neuronaux artificiels.

Proposition d'une nouvelle méthode de caractérisation

Dans ce chapitre, nous proposons une méthode pour caractériser des classes de chirotopes sous forme de “boules” dans un certain espace. Cette méthode peut également être qualifiée de méthode de classification supervisée puisque si l'on possède des chirotopes qui ne sont pas classés, on peut tester facilement à quelle boule ils appartiennent et ainsi classer ces chirotopes.

Pour cela, nous définirons d'abord deux types de $(d+1)$ -uplets qui font partie des $(d+1)$ -uplets non-fixes : les $(d+1)$ -uplets *caractéristiques* d'un sous-ensemble de chirotopes \mathcal{C} , qui sont les $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ tels que la plupart des chirotopes de \mathcal{C} aient le même signe, puis les $(d+1)$ -uplets *discriminantes* entre deux sous-ensembles de chirotopes \mathcal{C} et \mathcal{C}' , qui sont les $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ tels que la plupart des chirotopes de \mathcal{C} aient le même signe et que presque aucun chirotope de \mathcal{C}' n'aient ce signe.

Nous commençons par présenter notre méthode dans le cas uniforme puisqu'il s'agit du cas le plus simple et qui apparait le plus souvent lorsque l'on traite des données réelles. Cette approche a été présentée et illustré dans [23]. Nous présentons dans un deuxième temps notre méthode dans le cas non-uniforme et nous montrons qu'il s'agit d'une généralisation de la méthode dans le cas uniforme. Nous faisons ensuite le lien dans le cas non-uniforme entre les $(d+1)$ -uplets discriminantes et les $(d+1)$ -uplets caractéristiques et nous terminons ce chapitre en donnant un exemple dans le cas non-uniforme qui permet d'illustrer toute la méthode.

6.1 Cas uniforme

Soit \mathcal{M} un ensemble de chirotopes de même rang (noté r) définis sur le même ensemble abstrait \mathcal{E} . Pour tout sous-ensemble \mathcal{C} de \mathcal{M} , nous noterons $\overline{\mathcal{C}}$ le complémentaire de \mathcal{C} c'est-à-dire $\overline{\mathcal{C}} = \mathcal{M} \setminus \mathcal{C}$. Le cardinal d'un ensemble E (i.e. son nombre d'éléments) sera noté par $\#E$.

On rappelle que l'on note $\mathcal{B}(\mathcal{E})$ l'ensemble des plus petits $(d+1)$ -uplets de \mathcal{E} dans l'ordre lexicographique pour chaque classe de permutation (voir Section 2.1.1 pour plus de détails).

On rappelle que dans le cas uniforme, les chirotopes sont représentés par des points de $[-1, 1]^l$ où l est le nombre de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ (voir Section 5.1.2) et que chaque $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ forme une base.

 $\overline{\mathcal{C}}$
 $\#E$

Par abus de langage, on parlera du point moyen d'un sous-ensemble \mathcal{C} de \mathcal{M} lorsque nous évoquerons le point moyen de l'ensemble des points représentant un chirotope de \mathcal{C} .

Bases caractéristiques

Pour pouvoir discriminer une classe, c'est-à-dire trouver des propriétés partagées par tous les chirotopes de la classe de sorte que tout chirotope d'une autre classe ne vérifie pas ces propriétés, il faut commencer par chercher ce qui caractérise cette classe.

Soit B une base de $\mathcal{B}(\mathcal{E})$. On définit le *taux de caractérisation de la base B pour la classe \mathcal{C}* (noté $\theta_{\mathcal{C}}(B)$) par

$$\theta_{\mathcal{C}}(B) = \max \left\{ \frac{\#\{\chi \in \mathcal{C} \mid \chi(B) = +\}}{\#\mathcal{M}}, \frac{\#\{\chi \in \mathcal{C} \mid \chi(B) = -\}}{\#\mathcal{M}} \right\}$$

On dit qu'une base B est *totale-ment caractéristique pour la classe \mathcal{C}* si le signe de la base B est le même pour tout chirotope de \mathcal{C} . Le taux de caractérisation d'une base totalement caractéristique est donc égal à 1.

Caractérisation faible et bases discriminantes

Afin de trouver des sous-ensembles de bases permettant de discriminer les classes, nous allons généraliser la distance entre chirotopes définie dans la Section 2.3 à un sous-ensemble quelconque de bases. Soient M_1 et M_2 deux chirotopes de \mathcal{M} et χ_1 et χ_2 leur chirotope. Pour tout sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$, on définit la *distance entre M_1 et M_2 par rapport à \mathcal{D}* comme étant le nombre de bases de \mathcal{D} dont le signe est différent dans les chirotopes χ_1 et χ_2 . Formellement :

$$d_{\mathcal{D}}(M_1, M_2) = \#\{B \in \mathcal{D} \mid \chi_1(B) \neq \chi_2(B)\}$$

On définit de même pour un sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$ et deux points x et y de $\mathbb{R}^{\mathcal{B}(\mathcal{E})}$ la *distance entre x et y par rapport à \mathcal{D}* comme étant égale à :

$$d_{\mathcal{D}}(x, y) = \frac{\sum_{B \in \mathcal{D}} |x(B) - y(B)|}{2}$$

qui est égale à la moitié de la distance de Manhattan entre les projetés de ces points sur $\mathbb{R}^{\mathcal{D}}$. Dans ce qui suivra, nous utiliserons un abus de langage en disant que la distance $d_{\mathcal{D}}$ entre deux points est une distance dans $\mathbb{R}^{\mathcal{B}(\mathcal{E})}$ même quand $\mathcal{D} \neq \mathcal{B}(\mathcal{E})$.

Considérons maintenant une partition de \mathcal{M} en deux classes $\mathcal{M} = \mathcal{C} \uplus \overline{\mathcal{C}}$, ou plus généralement en k classes $\mathcal{M} = \mathcal{C}_1 \uplus \dots \uplus \mathcal{C}_k$. Une fois qu'une partition en classes est donnée, que ce soit par des experts ou par une méthode de classification automatique, nous cherchons un critère combinatoire qui soit simple et significatif afin de discriminer les classes.

Nous allons alors construire des classes dépendant de trois paramètres : un ensemble de bases \mathcal{D} , un point x de $\mathbb{R}^{\mathcal{B}(\mathcal{E})}$ et un réel r . On définit alors

$$\mathcal{I}_{\mathcal{D}, x, r} = \{\chi \in \mathcal{M} \mid d_{\mathcal{D}}(\chi, x) \leq r\}$$

On peut considérer cet ensemble comme étant l'ensemble des chirotopes contenus dans une boule de $\mathbb{R}^{\mathcal{B}(\mathcal{E})}$ centrée en x et de rayon r pour la distance $d_{\mathcal{D}}$.

taux de
caracté-
risation

base to-
talement
caractéris-
tique

$\mathcal{I}_{\mathcal{D}, x, r}$

En pratique, pour discriminer la classe \mathcal{C} de son complémentaire $\overline{\mathcal{C}}$, on cherche un sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$ le plus petit possible (pour plus de simplicité), un point x de $\mathbb{R}^{\mathcal{D}}$ à coordonnées dans $\{-1, 1\}$ (pour rester dans le cadre combinatoire) et un entier r tels que $\mathcal{I}_{\mathcal{D},x,l}$ soit égal à la classe \mathcal{C} . Donc un chirotope χ appartient à la classe \mathcal{C} si et seulement si il existe au plus r bases B de \mathcal{D} telles que $\chi(B) \neq x(B)$. On dit alors que $\mathcal{I}_{\mathcal{D},x,l}$ est une *caractérisation faible du couple* $(\mathcal{C}, \overline{\mathcal{C}})$ (par opposition à la caractérisation forte que nous verrons plus tard). Ainsi, \mathcal{D} correspond à un ensemble significatif de bases, x correspond aux valeurs prises par cet ensemble de bases pour la classe \mathcal{C} et r mesure combien de ces valeurs sont satisfaites par un chirotope.

caractérisation
faible

Nous allons illustrer cette notion avec un exemple simple dans lequel on considère que \mathcal{D} ne contient qu'une seule base B (i.e. $\mathcal{D} = \{B\}$). Alors x est donné par un seul signe, celui pour la base B . Sans perdre de généralité, on peut considérer que $x(B) = +1$. Ainsi un chirotope χ appartient à \mathcal{C} si et seulement si $\chi(B) = +1$ et χ appartient à $\overline{\mathcal{C}}$ si et seulement si $\chi(B) = -1$. Dans ce cas, on a $\mathcal{C} = \mathcal{I}_{\{B\},+1,0}$. On dit alors que B est *totalelement discriminante entre \mathcal{C} et $\overline{\mathcal{C}}$* . Ceci est la discrimination d'une classe la plus simple possible. Notons que si la partition de \mathcal{M} comporte k classes (i.e. $\mathcal{M} = \mathcal{C}_1 \uplus \dots \uplus \mathcal{C}_k$), et qu'une base B est totalelement discriminante entre \mathcal{C}_i et $\overline{\mathcal{C}_i}$ alors on peut déterminer uniquement à partir de B si un chirotope appartient à \mathcal{C}_i ou à son complémentaire $\overline{\mathcal{C}_i}$.

base to-
talement
discrimi-
nante

Remarque. Les notions de bases totalelement discriminantes et de bases totalelement caractéristiques sont liées. Plus précisément, si B est une base totalelement discriminante entre \mathcal{C} et $\overline{\mathcal{C}}$ alors \overline{B} est totalelement caractéristique pour la classe \mathcal{C} et B est totalelement caractéristique pour la classe $\overline{\mathcal{C}}$. En effet, pour tout chirotope χ_1 de \mathcal{C} et pour tout chirotope χ_2 de $\overline{\mathcal{C}}$, on a soit $\chi_1(B) = +$ et $\chi_2(B) = -$ soit $\chi_1(B) = -$ et $\chi_2(B) = +$. Ainsi $\theta_{\mathcal{C}}(B) = 1$ et $\theta_{\overline{\mathcal{C}}}(B) = 1$. B est donc totalelement caractéristique pour la classe \mathcal{C} et totalelement caractéristique pour la classe $\overline{\mathcal{C}}$. Pour que la réciproque soit vraie il faut rajouter comme condition que les signes des chirotopes de \mathcal{C} soient différents des signes des chirotopes de $\overline{\mathcal{C}}$.

Le problème pour discriminer la classe \mathcal{C} de son complémentaire $\overline{\mathcal{C}}$ est donc de déterminer les paramètres \mathcal{D} , x et r . Pour toute base B de $\mathcal{B}(\mathcal{E})$, on définit le *taux de discriminabilité de B entre \mathcal{C} et $\overline{\mathcal{C}}$* (noté $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$) par $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = |m_{\mathcal{C}}(B) - m_{\overline{\mathcal{C}}}(B)|/2$ où $m_{\mathcal{C}}$ et $m_{\overline{\mathcal{C}}}$ sont respectivement le point moyen de \mathcal{C} et le point moyen de $\overline{\mathcal{C}}$. Cette valeur appartient à $[0, 1]$. Plus cette valeur est proche de 1 et plus la base est significative pour discriminer \mathcal{C} de $\overline{\mathcal{C}}$.

taux de
discrimi-
nabilité

Le cas extrême est $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = 1$ ce qui implique que $m_{\mathcal{C}}(B) = +1$ et $m_{\overline{\mathcal{C}}}(B) = -1$ (ou $m_{\mathcal{C}}(B) = -1$ et $m_{\overline{\mathcal{C}}}(B) = +1$). Par définition du point moyen, on a alors que tout chirotope de \mathcal{C} (respectivement $\overline{\mathcal{C}}$) a comme signe $+$ (resp. $-$) pour la base B (ou $-$ et $+$ respectivement), ce qui signifie que B est une base totalelement discriminante entre \mathcal{C} et $\overline{\mathcal{C}}$.

Au contraire, $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = 0$ si la proportion de chirotopes de \mathcal{C} ayant comme signe $+$ pour la base B est égale à la proportion de chirotopes de $\overline{\mathcal{C}}$ ayant comme signe $+$ pour la base B . C'est le cas par exemple si la base B a le même signe pour tous les chirotopes de \mathcal{M} . Dans ce cas, la base B ne nous servira pas à discriminer \mathcal{C} de $\overline{\mathcal{C}}$.

Pour calculer les paramètres \mathcal{D} , x et r , nous allons trier les bases de $\mathcal{B}(\mathcal{E})$ en fonction du taux de discriminabilité. Comme $\mathcal{B}(\mathcal{E})$ peut contenir beaucoup d'éléments (en ne conservant que les bases non-fixes, nous avons près de 2 700 000 bases pour l'application que nous verrons dans la Section 11), il n'est pas possible de générer par un programme tous les sous-ensembles possibles

pour \mathcal{D} . Afin de réduire le nombre de sous-ensembles possibles pour \mathcal{D} , nous allons nous concentrer uniquement sur les bases qui ont le plus grand taux de discriminabilité car ce sont elles qui ont le plus de chance d'être significatives pour discriminer \mathcal{C} de $\overline{\mathcal{C}}$. Les bases ayant le plus grand taux de discriminabilité vont donc nous servir à construire \mathcal{D} et à déterminer la valeur de r . Pour déterminer x il nous suffit d'avoir les valeurs de $x(B)$ pour toute base B de \mathcal{D} . Ces dernières nous seront données par le signe de $m_{\mathcal{C}}(B)$ où $m_{\mathcal{C}}$ est le point moyen de \mathcal{C} .

La méthode pour déterminer les autres paramètres est expliquée plus en détail dans la Section 6.5.

Caractérisation forte

caractérisation
forte

Une *caractérisation forte du couple* $(\mathcal{C}, \overline{\mathcal{C}})$ dépend de quatre paramètres : un ensemble de bases \mathcal{D} , deux points x et \bar{x} de $\mathbb{R}^{\mathcal{B}(\mathcal{E})}$ et un réel r . Cette caractérisation forte consiste à trouver ces quatre paramètres de sorte que \mathcal{C} soit égale à $\mathcal{I}_{\mathcal{D},x,r}$ et que $\overline{\mathcal{C}}$ soit égale à $\mathcal{I}_{\mathcal{D},\bar{x},r}$. Comme $\mathcal{C} \cup \overline{\mathcal{C}} = \emptyset$, on a nécessairement que $r < \#\mathcal{D}/2$.

En pratique, tout comme pour une caractérisation faible, on cherche un sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$ le plus petit possible (pour plus de simplicité), un point x de $\mathbb{R}^{\mathcal{D}}$ à coordonnées dans $\{-1, 1\}$ et un entier $r < \#\mathcal{D}/2$. Le point \bar{x} s'obtient en prenant l'opposé de chaque coordonnée de x (c'est-à-dire pour tout $B \in \mathcal{D}$, $\bar{x}(B) = -x(B)$). Comme pour les caractérisations faibles, pour toute base B de \mathcal{D} , la valeur de $x(B)$ est égale au signe de $m_{\mathcal{C}}(B)$.

Lien entre caractérisation forte et caractérisation faible

Bien sûr, si $(\mathcal{I}_{\mathcal{D},x,r}, \mathcal{I}_{\mathcal{D},\bar{x},r})$ est une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$ (i.e. $\mathcal{C} = \mathcal{I}_{\mathcal{D},x,r}$ et $\overline{\mathcal{C}} = \mathcal{I}_{\mathcal{D},\bar{x},r}$) alors $\mathcal{I}_{\mathcal{D},x,r}$ est une caractérisation faible de $(\mathcal{C}, \overline{\mathcal{C}})$ et $\mathcal{I}_{\mathcal{D},\bar{x},r}$ est une caractérisation faible de $(\overline{\mathcal{C}}, \mathcal{C})$.

Si nous avons k classes telles que $\mathcal{M} = \mathcal{C}_1 \uplus \dots \uplus \mathcal{C}_k$ alors il nous faut déterminer les trois paramètres \mathcal{D} , x et r pour chaque classe \mathcal{C}_i . Une fois que nous avons tous les paramètres, nous avons deux façon de discriminer une classe \mathcal{C}_i . Cela vient du fait que si M appartient à \mathcal{C}_i alors on peut dire directement que $M \in \mathcal{C}_i$ ou dire de façon équivalente que $M \notin \mathcal{C}_j$ pour tout $j \neq i$. Ainsi pour discriminer \mathcal{C}_i on peut utiliser la méthode directe provenant des trois paramètres calculés pour \mathcal{C}_i ou la méthode indirecte qui provient de la négation des critères discriminants \mathcal{C}_j pour $j \neq i$.

6.2 Cas non-uniforme

Nous allons maintenant traiter le cas non-uniforme. Nous avons vu que le cas uniforme et le cas non-uniforme nécessitent des techniques différentes. Par exemple, pour représenter les chirotopes par des points (Section 5.1.2), les points appartiennent à \mathbb{R}^l dans le cas uniforme et à $\mathbb{R}^{3 \times l}$ dans le cas non uniforme (avec l étant le nombre de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$). Nous montrerons dans la section suivante que la méthode dans le cas non-uniforme est une généralisation de celle dans le cas uniforme.

6.2.1 Définitions et exemples

Soit S un sous-ensemble de $\{-, 0, +\}$. On note $\overline{S} = \{-, 0, +\} \setminus S$.

Soit \mathcal{M} un ensemble de chirotopes de même rang (noté r) définis sur le même ensemble abstrait \mathcal{E} et B un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$. On note $\eta_{\mathcal{M}}(B, S)$ le nombre de chirotopes de \mathcal{M} dont le signe de B appartient à S et $\rho_{\mathcal{M}}(B, S)$ la proportion de chirotopes de \mathcal{M} dont le signe de B appartient à S . Précisément :

$$\begin{aligned}\eta_{\mathcal{M}}(B, S) &= \#\{M \in \mathcal{M} \mid \chi_M(B) \in S\} \\ \rho_{\mathcal{M}}(B, S) &= \frac{\eta_{\mathcal{M}}(B, S)}{|\mathcal{M}|}\end{aligned}$$

$$\begin{aligned}\eta_{\mathcal{M}}(B, S) \\ \rho_{\mathcal{M}}(B, S)\end{aligned}$$

Propriété 6.1. Soient B un $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ et S un sous-ensemble de $\{-, 0, +\}$. On a alors

$$\eta_{\mathcal{M}}(B, S) = \sum_{\{s\} \in S} \eta_{\mathcal{M}}(B, \{s\}) \quad \text{et} \quad \rho_{\mathcal{M}}(B, S) = \sum_{\{s\} \in S} \rho_{\mathcal{M}}(B, \{s\})$$

On a également que pour tout $S' \subseteq S$

$$\rho_{\mathcal{M}}(B, S') \leq \rho_{\mathcal{M}}(B, S)$$

On a de plus

$$1/3 \leq \max_{s \in \{-, 0, +\}} \rho_{\mathcal{M}}(B, \{s\}) \leq 1$$

Propriété 6.2. Soit S un sous-ensemble de $\{-, 0, +\}$. On a alors

$$\rho_{\mathcal{M}}(B, S) = 1 - \rho_{\mathcal{M}}(B, \bar{S})$$

Nous allons maintenant définir les $(d+1)$ -uplets caractéristiques puis les $(d+1)$ -uplets discriminants dans le cas non-uniforme.

Les $(d+1)$ -uplets caractéristiques

Soit B un $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$. On définit le *taux de caractérisation de B pour \mathcal{M}* (noté $\theta_{\mathcal{M}}(B)$) de la façon suivante :

$$\theta_{\mathcal{M}}(B) = \max_{\substack{S \subseteq \{-, 0, +\} \\ S \neq \{-, 0, +\}}} \rho_{\mathcal{M}}(B, S)$$

taux de caractérisation

Soit S un sous-ensemble strict de $\{-, 0, +\}$. On dit que le $(d+1)$ -uplet B est *totalement caractéristique de \mathcal{M} pour l'ensemble de signes S* à un taux $\rho_{\mathcal{M}}(B, S)$ si le signe du $(d+1)$ -uplet B appartient à S pour tout chirotope de \mathcal{M} . Le taux de caractérisation d'un $(d+1)$ -uplet totalement caractéristique est donc égal à 1.

$(d+1)$ -uplet totalement caractéristique

Soit S un sous-ensemble strict de $\{-, 0, +\}$. On dit que $\theta_{\mathcal{M}}(B)$ est atteint pour l'ensemble de signes S si $\theta_{\mathcal{M}}(B) = \rho_{\mathcal{M}}(B, S)$.

$\theta_{\mathcal{M}}(B)$ est atteint pour S

Le taux de caractérisation nous permet de créer une hiérarchie au sein des $(d+1)$ -uplets caractéristiques : on dit qu'un $(d+1)$ -uplet B est un $(d+1)$ -uplet plus caractéristique de \mathcal{M} qu'un $(d+1)$ -uplet B' si $\theta_{\mathcal{M}}(B) > \theta_{\mathcal{M}}(B')$. Cela nous permet de comparer les $(d+1)$ -uplets caractéristiques entre eux pour par exemple ne faire des calculs que sur les $(d+1)$ -uplets les plus caractéristiques.

Exemple. Nous allons illustrer toutes les notions de cette section grâce à cet exemple. Considérons un ensemble \mathcal{M} de 200 chirotopes définis sur le même ensemble abstrait \mathcal{E} et un sous-ensemble \mathcal{C} de \mathcal{M} de taille 50. Soit B un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ tel que le nombre de chirotopes en fonction du signe de B soit donné par le Tableau 6.1. Le pourcentage de chirotopes est alors donné par le Tableau 6.2.

s	-	0	+
$\eta_{\mathcal{C}}(B, \{s\})$	25	25	0
$\eta_{\overline{\mathcal{C}}}(B, \{s\})$	0	0	150
$\eta_{\mathcal{M}}(B, \{s\})$	25	25	150

TABLE 6.1 – Répartition des chirotopes en fonction des signes du $(d+1)$ -uplet B

S	$\{-\}$	$\{0\}$	$\{+\}$	$\{-, 0\}$	$\{0, +\}$	$\{-, +\}$
$\rho_{\mathcal{C}}(B, S)$	50%	50%	0%	100%	50%	50%
$\rho_{\overline{\mathcal{C}}}(B, S)$	0%	0%	100%	0%	100%	100%
$\rho_{\mathcal{M}}(B, S)$	12,5%	12,5%	75%	35%	87,5%	87,5%

TABLE 6.2 – Valeurs de $\rho_{\mathcal{C}}(B, S)$, $\rho_{\overline{\mathcal{C}}}(B, S)$ et $\rho_{\mathcal{M}}(B, S)$ en fonction de l'ensemble de signes S

B est alors un $(d+1)$ -uplet totalement caractéristique de \mathcal{C} et de $\overline{\mathcal{C}}$ respectivement pour $\{-, 0\}$ et $\{+\}$. Par contre le taux de caractérisation de B pour \mathcal{M} est de 87,5%. Ce taux est atteint pour $\{0, +\}$ et pour $\{-, +\}$.

Les $(d+1)$ -uplets discriminants

Soit \mathcal{C} un sous-ensemble de \mathcal{M} . On cherche un sous-ensemble de bases permettant de discriminer \mathcal{C} de $\overline{\mathcal{C}}$.

Tout comme dans le cas uniforme, on souhaite savoir pour tout $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ si B permet de séparer \mathcal{C} de $\overline{\mathcal{C}}$. Or dans le cas uniforme, si une base B permet de séparer \mathcal{C} de $\overline{\mathcal{C}}$, il n'y a qu'un seul cas de figure : l'un de ces deux sous-ensembles contient tous les chirotopes qui ont pour signe + pour la base B et l'autre sous-ensemble contient tous les chirotopes qui ont pour signe - pour la base B . Dans le cas non-uniforme, on ne peut pas savoir à l'avance de quels chirotopes sont composés les deux sous-ensembles. En effet, dans le cas non-uniforme il y a trois cas de figures : un des sous-ensembles contient soit tous les chirotopes qui ont pour signe + (et donc l'autre sous-ensemble contient tous les chirotopes qui ont pour signe 0 ou -), soit tous les chirotopes qui ont pour signe 0, soit tous les chirotopes qui ont pour signe -. Pour prendre en compte les différents cas, on introduit alors la mesure suivante pour tout sous-ensemble strict S de $\{+, 0, -\}$:

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = \frac{\rho_{\mathcal{C}}(B, S) + \rho_{\overline{\mathcal{C}}}(B, \overline{S}) - \rho_{\mathcal{C}}(B, \overline{S}) - \rho_{\overline{\mathcal{C}}}(B, S)}{2}$$

Remarque. Soit S un sous-ensemble strict de $\{+, 0, -\}$. Plaçons nous dans le cas où nous souhaitons déterminer si les chirotopes de \mathcal{M} appartiennent à \mathcal{C} ou à $\overline{\mathcal{C}}$ en fonction des signes d'un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$. Nous allons alors partitionner l'ensemble \mathcal{M} de chirotopes en deux ensembles \mathcal{M}_S et $\overline{\mathcal{M}}_S$ tels que \mathcal{M}_S contienne les chirotopes dont le signe de B appartient à S et que $\overline{\mathcal{M}}_S$ contienne les autres chirotopes (i.e. les chirotopes dont le signe de B appartient à \overline{S}). $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$ nous indique alors sur l'erreur que l'on fait lorsque l'on suppose que $\mathcal{M}_S = \mathcal{C}$ et que $\overline{\mathcal{M}}_S = \overline{\mathcal{C}}$. En effet, $\rho_{\mathcal{C}}(B, S)$ correspond à la proportion de chirotopes de \mathcal{C} qui appartiennent à \mathcal{M}_S . De même $\rho_{\overline{\mathcal{C}}}(B, \overline{S})$ correspond à la proportion de chirotopes de $\overline{\mathcal{C}}$ qui appartiennent à $\overline{\mathcal{M}}_S$. $\rho_{\mathcal{C}}(B, \overline{S})$ et $\rho_{\overline{\mathcal{C}}}(B, S)$ correspondent par contre au nombre de chirotopes qui seraient mal classés puisqu'il s'agit respectivement de la proportion de chirotopes de \mathcal{C} qui n'appartiennent pas à \mathcal{M}_S et de la proportion de chirotopes de $\overline{\mathcal{C}}$ qui n'appartiennent pas à $\overline{\mathcal{M}}_S$.

Ainsi si $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = 1$ cela signifie que $\mathcal{M}_S = \mathcal{C}$ et que $\overline{\mathcal{M}}_S = \overline{\mathcal{C}}$. A l'inverse si $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = -1$ cela signifie que $\mathcal{M}_S = \overline{\mathcal{C}}$ et que $\overline{\mathcal{M}}_S = \mathcal{C}$.

Si par contre $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = 0$ cela indique que le couple (B, S) ne nous permet pas de répartir les chirotopes de \mathcal{M} dans les sous-ensembles \mathcal{C} et $\overline{\mathcal{C}}$.

Observation. Comme $\rho_{\mathcal{C}}(B, S) = 1 - \rho_{\mathcal{C}}(B, \overline{S})$, on a

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = \rho_{\mathcal{C}}(B, S) - \rho_{\overline{\mathcal{C}}}(B, S)$$

Comme nous l'avons vu au Chapitre 5.1.2, dans le cas non-uniforme on associe à chaque chirotope χ un point p de $[0, 1]^{3 \times l}$ où $l = \binom{n}{d+1}$. Pour chaque $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ on fait correspondre trois coordonnées (que l'on note B_- , B_0 et B_+) telles que pour $s = \chi(B)$ on a $p(B_s) = 1$ et les valeurs de p pour les deux autres coordonnées sont nulles. Si l'on considère l'ensemble des points associés aux chirotopes de \mathcal{C} , alors $\rho_{\mathcal{C}}(B, \{s\})$ correspond à la valeur du point moyen de \mathcal{C} pour la coordonnée B_s . Ainsi en notant $m_{\mathcal{C}}$ le point moyen de \mathcal{C} on a :

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\}) = m_{\mathcal{C}}(B_s) - m_{\overline{\mathcal{C}}}(B_s).$$

Observation. On a les relations suivantes :

$$\begin{aligned} \tau(B, \overline{\mathcal{C}}, \mathcal{C}, S) &= -\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) \\ \tau(B, \mathcal{C}, \overline{\mathcal{C}}, \overline{S}) &= -\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) \end{aligned}$$

On en déduit plusieurs propriétés :

Propriété 6.3.

$$\sum_{\substack{S \subset \{-, 0, +\} \\ S \neq \{-, 0, +\}}} \tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = 0$$

Preuve.

$$\sum_{\substack{S \subset \{-, 0, +\} \\ S \neq \{-, 0, +\}}} \tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = \sum_{s \in \{-, 0, +\}} (\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\}) - \tau(B, \mathcal{C}, \overline{\mathcal{C}}, \overline{\{s\}}))$$

On obtient ainsi le résultat par la relation précédente. \square

Propriété 6.4.

$$\sum_{s \in \{-, 0, +\}} \tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\}) = 0$$

Preuve.

$$\begin{aligned} \sum_{s \in \{-, 0, +\}} \tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\}) &= \sum_{s \in \{-, 0, +\}} (\rho_{\mathcal{C}}(B, \{s\}) - \rho_{\overline{\mathcal{C}}}(B, \{s\})) \\ &= \sum_{s \in \{-, 0, +\}} \rho_{\mathcal{C}}(B, \{s\}) - \sum_{s \in \{-, 0, +\}} \rho_{\overline{\mathcal{C}}}(B, \{s\}) \end{aligned}$$

Par définition, $\sum_{s \in \{-, 0, +\}} \rho_{\mathcal{C}}(B, \{s\}) = 1$ ce qui prouve le résultat. \square

Remarque. Grâce aux propriétés précédentes il nous suffit de calculer les valeurs de $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$ et de $\tau(B, \overline{\mathcal{C}}, \mathcal{C}, S)$ pour les ensembles S de taille 1. S'il n'y a pas d'ambiguïté, pour alléger les notations nous noterons $\{+\}$, $\{0\}$ et $\{-\}$ respectivement par $+$, 0 et $-$.

taux de discriminabilité

Soit B un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$. On définit le *taux de discriminabilité de B entre \mathcal{C} et $\overline{\mathcal{C}}$* (noté $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$) de la façon suivante :

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = \max_{\substack{S \subset \{-, 0, +\} \\ S \neq \{-, 0, +\}}} \tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$$

$(d+1)$ -uplet totalement discriminant

On dit que un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ est un $(d+1)$ -uplet *totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$* s'il existe un sous-ensemble S de $\{-, 0, +\}$ tel que le signe de B appartienne à S pour tous les chirotopes de \mathcal{C} et que le signe de B appartienne à \overline{S} pour tous les chirotopes de $\overline{\mathcal{C}}$. Ainsi si B est un $(d+1)$ -uplet totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$, alors $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = 1$.

Comme pour les $(d+1)$ -uplets caractéristiques, on crée une hiérarchie dans les $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ via le taux de discriminabilité : un $(d+1)$ -uplet B est plus discriminante qu'un $(d+1)$ -uplet B' si $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) > \tau(B', \mathcal{C}, \overline{\mathcal{C}})$.

$\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour S

Soit S un sous-ensemble strict de $\{-, 0, +\}$. On dit que $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour l'ensemble de signes S si $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = \tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$. Ainsi si un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ est totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$ et que $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour un certain sous-ensemble strict S de $\{-, 0, +\}$ (i.e. $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = 1$), alors un chirotipe χ appartient à \mathcal{C} si et seulement si $\chi(B) \in S$ et χ appartient à $\overline{\mathcal{C}}$ si et seulement si $\chi(B) \in \overline{S}$.

D'après les propriétés de $\tau(B, \overline{\mathcal{C}}, \mathcal{C}, S)$ on obtient la propriété suivante :

Propriété 6.5.

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = \tau(B, \overline{\mathcal{C}}, \mathcal{C})$$

Preuve. Notons T l'ensemble de signes tel que $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour T . On a alors

$$\begin{aligned} \tau(B, \mathcal{C}, \overline{\mathcal{C}}) &= \tau(B, \mathcal{C}, \overline{\mathcal{C}}, T) \\ &= -\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \overline{T}) \\ &= \tau(B, \overline{\mathcal{C}}, \mathcal{C}, \overline{T}) \\ &= \tau(B, \overline{\mathcal{C}}, \mathcal{C}) \end{aligned}$$

\square

Tout comme pour les calculs de $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$, on peut obtenir la valeur du taux de discriminabilité en ne considérant que les ensembles de signes de taille 1. On a alors :

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = \max_{s \in \{-, 0, +\}} \left\{ \max \left\{ \tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\}), \tau(B, \overline{\mathcal{C}}, \mathcal{C}, \{s\}) \right\} \right\}$$

S	$\{-\}$	$\{0\}$	$\{+\}$	$\{-, 0\}$	$\{0, +\}$	$\{-, +\}$
$\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$	0,5	0,5	-1	1	-0,5	-0,5

TABLE 6.3 – Valeurs de $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$ en fonction de S

s	$-$	0	$+$
$\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\})$	0,5	0,5	-1
$\tau(B, \overline{\mathcal{C}}, \mathcal{C}, \{s\})$	-0,5	-0,5	1

TABLE 6.4 – Valeurs de $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{s\})$ et de $\tau(B, \overline{\mathcal{C}}, \mathcal{C}, \{s\})$ en fonction de s

Exemple. Poursuivons l'étude de l'exemple précédent. Par définition de $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$, $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est la valeur maximale du Tableau 6.3. Ainsi $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = 100\%$. B est donc un $(d+1)$ -uplet totalement discriminante entre \mathcal{C} et $\overline{\mathcal{C}}$. Le taux de discriminabilité de B entre \mathcal{C} et $\overline{\mathcal{C}}$ vaut 1 et est atteint pour l'ensemble de signes $\{-, 0\}$. Cela signifie que si l'on classe les chirotopes de \mathcal{M} en deux classes : ceux qui ont comme signe $-$ ou 0 pour le $(d+1)$ -uplet B et ceux qui ont comme signe $+$ pour B , alors ces deux classes sont égales respectivement à \mathcal{C} et à $\overline{\mathcal{C}}$.

On arrive à la même conclusion en n'utilisant que le Tableau 6.4. En effet on remarque que dans ce tableau que la valeur 1 est atteinte par $\tau(B, \overline{\mathcal{C}}, \mathcal{C}, \{+\})$. Cela signifie que tous les chirotopes de $\overline{\mathcal{C}}$ ont comme signe $+$ pour le $(d+1)$ -uplet B et que tous les chirotopes de \mathcal{C} ont comme signe 0 ou $-$ pour le $(d+1)$ -uplet B .

6.2.2 Caractérisations faible et forte des classes

Si un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ est totalement discriminant entre un sous-ensemble de chirotopes \mathcal{C} et son complémentaire $\overline{\mathcal{C}}$, alors B est suffisant pour déterminer si un chirotope appartient à \mathcal{C} ou à $\overline{\mathcal{C}}$. En effet, nous avons vu que si un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ est totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$ et que $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour un certain sous-ensemble strict S de $\{-, 0, +\}$ (i.e. $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = 1$), alors un chirotope χ appartient à \mathcal{C} si et seulement si $\chi(B) \in S$ et χ appartient à $\overline{\mathcal{C}}$ si et seulement si $\chi(B) \in \overline{S}$.

Or si aucune base n'est totalement discriminante nous allons utiliser plusieurs $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$ pour classer les chirotopes. D'après ce que nous venons de voir, lorsque l'on caractérise une classe \mathcal{C} et son complémentaire $\overline{\mathcal{C}}$ par un seul $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$, il est important de connaître

un sous-ensemble strict S de $\{-, 0, +\}$ tel que le taux de discriminabilité de B entre \mathcal{C} et $\overline{\mathcal{C}}$ soit atteint pour S . Il nous faut donc généraliser cela pour un plus grand nombre de $(d+1)$ -uplets.

Pour tout $\mathcal{D} \subseteq \mathcal{B}(\mathcal{E})$ et pour tout couple $(\mathcal{C}, \overline{\mathcal{C}})$, nous allons construire une liste ordonnée de taille $\#\mathcal{D}$ qui contiendra un sous-ensemble strict de $\{-, 0, +\}$ pour chaque $(d+1)$ -uplet de \mathcal{D} . Pour cela, on commence par ordonner les $(d+1)$ -uplets de \mathcal{D} selon le taux de discriminabilité par ordre décroissant. On note alors $\mathcal{D} = (B_1, \dots, B_p)$ où p est le cardinal de \mathcal{D} . Pour tout $i \in \{1, \dots, p\}$, parmi tous les sous-ensembles stricts de $\{-, 0, +\}$ tels que $\tau(B_i, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour ces sous-ensembles, on en choisit un qui est minimal pour l'inclusion (c'est-à-dire que si $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour $\{s\}$ et $\{s, t\}$ avec $s, t \in \{-, 0, +\}$, on choisit le sous-ensemble $\{s\}$). On ajoute alors le sous-ensemble que nous venons de choisir à la liste que nous sommes en train de construire. Ainsi le i -ème élément de la liste est un sous-ensemble strict de $\{-, 0, +\}$ tel que $\tau(B_i, \mathcal{C}, \overline{\mathcal{C}})$ est atteint pour ce sous-ensemble.

Lorsque l'on choisit un sous-ensemble de signe minimal pour l'inclusion et qu'on le rajoute à la liste, il se peut que l'on doive choisir parmi plusieurs sous-ensembles. On note $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$ l'ensemble de toutes les listes que l'on peut générer de cette manière.

Caractérisation faible

A partir des classes \mathcal{C} et $\overline{\mathcal{C}}$, nous allons créer une classe dépendant de trois paramètres : un sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$, une liste T de taille $\#\mathcal{D}$ contenant des sous-ensembles stricts de $\{-, 0, +\}$ et un entier r . Nous noterons cette classe $\mathcal{I}_{\mathcal{D}, T, q}$. En pratique T sera une liste de $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$.

A chaque $(d+1)$ -uplet B de \mathcal{D} correspond dans la liste T un sous-ensemble strict de $\{-, 0, +\}$. Plus précisément, en ordonnant \mathcal{D} selon le taux de discriminabilité par ordre décroissant, le i -ème $(d+1)$ -uplet de \mathcal{D} correspond au i -ème élément de la liste T . Pour tout $(d+1)$ -uplet B de \mathcal{D} , on note alors $T(B)$ le sous-ensemble strict de $\{-, 0, +\}$ correspondant à B dans la liste T . Pour tout chirotope χ de \mathcal{M} , on compte le nombre de $(d+1)$ -uplets B de \mathcal{D} , tels que le signe de B dans χ appartienne au sous-ensemble de signes $T(B)$.

On définit $\mathcal{I}_{\mathcal{D}, T, q}$ comme étant l'ensemble des chirotopes de \mathcal{M} pour lesquels ce nombre est supérieur ou égal à q . Formellement on a :

$$\chi \in \mathcal{I}_{\mathcal{D}, T, q} \iff \#\{B \in \mathcal{D} \mid \chi(B) \in T(B)\} \geq q$$

Ainsi, pour discriminer la classe \mathcal{C} de son complémentaire $\overline{\mathcal{C}}$, on cherche un sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$ le plus petit possible (pour plus de simplicité), une liste T de $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$ et un entier q tels que $\mathcal{I}_{\mathcal{D}, T, q}$ soit égal à \mathcal{C} . On dit alors que $\mathcal{I}_{\mathcal{D}, T, q}$ est une *caractérisation faible* du couple $(\mathcal{C}, \overline{\mathcal{C}})$. La méthode pour déterminer ces paramètres est expliqué plus en détail dans la Section 6.5.

Caractérisation forte

Une *caractérisation forte* du couple $(\mathcal{C}, \overline{\mathcal{C}})$ dépend de quatre paramètres : un ensemble \mathcal{D} de $(d+1)$ -uplets de $\mathcal{B}(\mathcal{E})$, deux listes T et \overline{T} de taille $\#\mathcal{D}$ contenant des sous-ensembles stricts de $\{-, 0, +\}$ et un entier q . Cette caractérisation forte consiste à trouver ces quatre paramètres de sorte que \mathcal{C} soit égale à $\mathcal{I}_{\mathcal{D}, T, q}$ et que $\overline{\mathcal{C}}$ soit égale à $\mathcal{I}_{\mathcal{D}, \overline{T}, q}$. Comme $\mathcal{C} \cup \overline{\mathcal{C}} = \mathcal{B}(\mathcal{E})$, on a nécessairement que $\#\mathcal{D}/2 < q \leq \#\mathcal{D}$.

Ainsi, si les valeurs de ces paramètres ne permettent pas d'avoir une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$ (c'est-à-dire si $\mathcal{C} \neq \mathcal{I}_{\mathcal{D}, T, q}$ et que $\overline{\mathcal{C}} \neq \mathcal{I}_{\mathcal{D}, \overline{T}, q}$), selon les valeurs de q il se peut qu'un certain nombre de chirotopes de \mathcal{M} n'appartiennent ni à $\mathcal{I}_{\mathcal{D}, T, q}$ ni à $\mathcal{I}_{\mathcal{D}, \overline{T}, q}$.

En pratique, tout comme pour une caractérisation faible, on cherche un sous-ensemble \mathcal{D} de $\mathcal{B}(\mathcal{E})$ le plus petit possible (pour plus de simplicité), une liste T de $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$ et un entier q tel que $\#\mathcal{D}/2 < q \leq \#\mathcal{D}$. La liste \overline{T} s'obtient en prenant pour chaque $(d+1)$ -uplet B de \mathcal{D} le complémentaire de $T(B)$ (c'est-à-dire pour tout $B \in \mathcal{D}$, $\overline{T}(B) = \{-, 0, +\} \setminus T(B)$).

Nous verrons dans la Section 6.5 comment se calculent les paramètres des caractérisations faibles et fortes.

Lien entre caractérisation forte et caractérisation faible

Tout comme dans le cas uniforme, si $(\mathcal{I}_{\mathcal{D}, T, q}, \mathcal{I}_{\mathcal{D}, \overline{T}, q})$ est une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$ (i.e. $\mathcal{C} = \mathcal{I}_{\mathcal{D}, T, q}$ et $\overline{\mathcal{C}} = \mathcal{I}_{\mathcal{D}, \overline{T}, q}$) alors $\mathcal{I}_{\mathcal{D}, T, q}$ est une caractérisation faible de $(\mathcal{C}, \overline{\mathcal{C}})$ et $\mathcal{I}_{\mathcal{D}, \overline{T}, q}$ est une caractérisation faible de $(\overline{\mathcal{C}}, \mathcal{C})$.

Cas d'une partition en plusieurs classes

Nous avons travaillé jusqu'à présent avec une partition de taille 2 de l'ensemble de chirotopes \mathcal{M} . Nous allons voir ce qui se passe si nous avons une partition de \mathcal{M} en un nombre c de classes.

Soit $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_c$ une partition de \mathcal{M} . Nous pouvons utiliser une caractérisation (faible ou forte) expliquée précédemment pour séparer \mathcal{C}_i et $\overline{\mathcal{C}_i}$ pour tout $i \in \{1, \dots, c\}$. Ainsi nous avons deux critères indépendants pour décrire chaque classe \mathcal{C}_i : le critère direct qui apparait quand on réalise une caractérisation (faible ou forte) entre \mathcal{C}_i et $\overline{\mathcal{C}_i}$ et le critère indirect qui est la négation des critères directs des classes \mathcal{C}_j pour $j \neq i$. En effet, comme $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_c$ est une partition de \mathcal{M} , on peut décrire le fait qu'un chirotope χ appartient à une classe \mathcal{C}_i en disant que χ n'appartient pas aux classes \mathcal{C}_j pour $j \neq i$.

6.3 Retour sur le cas uniforme

Comme nous l'avons dit précédemment, dans le cas uniforme si une base B permet de séparer \mathcal{C} de $\overline{\mathcal{C}}$, il n'y a qu'un seul cas de figure : l'un de ces deux sous-ensembles contient tous les chirotopes qui ont pour signe + pour la base B et l'autre sous-ensembles contient tous les chirotopes qui ont pour signe - pour la base B . Par contre dans le cas non-uniforme, si un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ permet de séparer le sous-ensemble \mathcal{C} du sous-ensemble $\overline{\mathcal{C}}$, on ne peut pas savoir à l'avance de quels chirotopes sont composés ces deux sous-ensembles. En effet, dans le cas non-uniforme il y a trois cas de figures : un des sous-ensembles contient soit tous les chirotopes qui ont pour signe + (et donc l'autre sous-ensemble contient tous les chirotopes qui ont pour signe 0 ou -), soit tous les chirotopes qui ont pour signe 0, soit tous les chirotopes qui ont pour signe -.

Ainsi dans le cas non-uniforme, que ce soit pour caractériser une classe ou pour la discriminer nous avons besoin d'utiliser les sous-ensembles de $\{+, 0, -\}$ de taille 1 et 2. Dans le cas uniforme, les seules valeurs sont + et -. Si l'on veut utiliser les mêmes définitions dans le cas uniforme que dans

le cas non-uniforme, il faut alors remplacer $\{+, 0, -\}$ par $\{+, -\}$. Les formulations “sous-ensemble strict S de $\{+, 0, -\}$ ” deviennent alors “sous-ensemble strict S de $\{+, -\}$ ”. Dans les définitions pour le cas non-uniforme, chaque fois que l’on utilise cette formulation, il n’y a pas un grand intérêt de prendre l’ensemble vide pour S . En utilisant les mêmes définitions pour le cas uniforme, les valeurs intéressantes que peut prendre S sont donc $\{+\}$ et $\{-\}$.

Taux de caractérisation et de discriminabilité. Nous allons détailler les définitions posées dans l’explication de la méthode pour le cas non-uniforme et montrer qu’elles s’appliquent au cas uniforme.

Il est facile de voir qu’en appliquant au cas uniforme les définitions de $(d+1)$ -uplets totalement caractéristiques et totalement discriminants de la Section 6.2, on obtient des définitions équivalentes à celles de la Section 6.1. En effet, si le signe de B est le même pour tout les chirotopes, ce signe est soit $+$ soit $-$ ce qui signifie bien qu’il existe un ensemble qui est soit égal à $\{+\}$, soit à $\{-\}$, qui contient ce signe.

En appliquant au cas uniforme la définition du taux de caractérisation de B pour \mathcal{M} de la Section 6.2, on obtient $\theta_{\mathcal{M}}(B) = \max\{\rho_{\mathcal{M}}(B, \{+\}), \rho_{\mathcal{M}}(B, \{-\})\}$. D’après la définition de $\rho_{\mathcal{M}}(B, S)$, cette définition du taux de caractérisation est bien équivalente à celle de la Section 6.1.

Dans le cas uniforme, on a :

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{+\}) = \frac{\rho_{\mathcal{C}}(B, \{+\}) + \rho_{\overline{\mathcal{C}}}(\overline{B}, \{-\}) - \rho_{\mathcal{C}}(B, \{-\}) - \rho_{\overline{\mathcal{C}}}(\overline{B}, \{+\})}{2}$$

Or en notant $m_{\mathcal{C}}$ le point moyen de \mathcal{C} , on a $m_{\mathcal{C}}(B) = (\rho_{\mathcal{C}}(B, \{+\}) - \rho_{\mathcal{C}}(B, \{-\}))/2$. Ainsi $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{+\}) = m_{\mathcal{C}}(B) - m_{\overline{\mathcal{C}}}(\overline{B})$. Or nous avons vu que $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = -\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \overline{S})$. Ainsi $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{-\}) = -\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{+\})$. Le maximum de ces deux valeurs est bien égale à la valeur absolue de $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, \{+\})$, donc à $|m_{\mathcal{C}}(B) - m_{\overline{\mathcal{C}}}(\overline{B})|$, ce qui montre bien l’équivalence des définitions du taux de discriminabilité dans le cas uniforme.

Caractérisations faibles et fortes. Nous allons maintenant comparer les caractérisations faibles et fortes de la Section 6.1 à celles de la Section 6.2.

Dans le cas uniforme (Section 6.1), les caractérisations faibles et fortes se basent sur des ensembles de chirotopes $\mathcal{I}_{\mathcal{D}, x, r}$ définis comme ceci :

$$\mathcal{I}_{\mathcal{D}, x, r} = \{\chi \in \mathcal{M} \mid d_{\mathcal{D}}(\chi, x) \leq r\}$$

où \mathcal{D} est un ensemble de bases de $\mathcal{B}(\mathcal{E})$, x est un point de $\mathbb{R}^{\mathcal{B}}$ et r est un réel r .

Dans le cas non-uniforme (Section 6.2.2), les caractérisations faibles et fortes se basent sur des ensembles de chirotopes $\mathcal{I}_{\mathcal{D}, T, q}$ tels que

$$\chi \in \mathcal{I}_{\mathcal{D}, T, q} \iff \#\{B \in \mathcal{D} \mid \chi(B) \in T(B)\} \geq q$$

où \mathcal{D} est un ensemble de $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$, T est une liste de sous-ensemble strict de $\{-, 0, +\}$, et q est un entier.

Nous allons voir que dans le cas uniforme, tout ensemble de chirotopes $\mathcal{I}_{\mathcal{D},T,q}$ est égal à un ensemble $\mathcal{I}_{\mathcal{D},x,r}$ pour un certain point x de $\mathbb{R}^{\mathcal{B}}$ et un certain réel r . Dans le cas non-uniforme, pour tout $(d+1)$ -uplet B de \mathcal{D} , $T(B)$ est un sous-ensemble strict de $\{+, 0, -\}$. Ainsi dans le cas uniforme $T(B)$ vaut soit $\{+\}$ soit $\{-\}$. On note alors x le point de $\mathbb{R}^{\mathcal{B}}$ tel que $x(B) = 0$ si $B \notin \mathcal{D}$, $x(B) = 1$ si $T(B) = \{+\}$ et $x(B) = -1$ si $T(B) = \{-\}$. Ainsi, pour tout chirotope χ de \mathcal{M} et toute base B de \mathcal{D} , $d_{\{B\}}(\chi, x) = 0$ si et seulement si $\chi(B) \in T(B)$. On a donc $\#\{B \in \mathcal{D} \mid \chi(B) \in T(B)\} = \#\mathcal{D} - d_{\mathcal{D}}(\chi, x)$. En posant $r = \#\mathcal{D} - q$ on a que $\mathcal{I}_{\mathcal{D},T,q}$ est égale à boule de $\mathbb{R}^{\mathcal{B}}$, de centre x et de rayon r pour la distance $d_{\mathcal{D}}$, c'est-à-dire $\mathcal{I}_{\mathcal{D},T,q} = \mathcal{I}_{\mathcal{D},x,r}$.

Nous venons de voir que l'ensemble de chirotopes $\mathcal{I}_{\mathcal{D},T,q}$ défini dans la Section 6.2 est une généralisation de la boule de $\mathbb{R}^{\mathcal{B}}$ définie dans la Section 6.1 (notée $\mathcal{I}_{\mathcal{D},x,r}$). Ainsi la méthode présentée dans la Section 6.2 est une généralisation de celle présentée dans la Section 6.1.

Cas de matroïdes non-orientés. A l'inverse du cas uniforme, dans le cas de matroïdes non-orientés, nous cherchons à déterminer si une base a pour signe 0 ou non. Dans ce cas pour toute base B , $\chi(B) \in \{0, \text{non} - 0\}$. On a alors $\rho_{\mathcal{M}}(B, \{\text{non} - 0\}) = 1 - \rho_{\mathcal{M}}(B, \{0\})$.

Les cas uniformes et non-orientés sont similaires puisque à chaque fois il n'y a que deux possibilités pour le signe d'un ensemble de points. Ainsi, comme toute la Section 6.2 est valable pour le cas uniforme, cette section peut également s'appliquer au cas non-orienté en remplaçant $\{-, +\}$ par $\{0, \text{non} - 0\}$.

6.4 Liens entre les taux de caractérisation et de discriminabilité

Maintenant que nous avons vu que les définitions de la Section 6.2 sont équivalentes à celles de la Section 6.1, nous allons voir le lien entre les notions de $(d+1)$ -uplets caractéristiques et de $(d+1)$ -uplets discriminants.

Les notions de $(d+1)$ -uplets totalement discriminants et de $(d+1)$ -uplets totalement caractéristiques sont liées. Plus précisément, si B est un $(d+1)$ -uplet totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$ alors B est totalement caractéristique pour la classe \mathcal{C} et B est totalement caractéristique pour la classe $\overline{\mathcal{C}}$. En effet comme B est totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$, pour tout chirotope χ_1 de \mathcal{C} et pour tout chirotope χ_2 de $\overline{\mathcal{C}}$, il existe un sous-ensemble S strict de $\{-, 0, +\}$ tel que $\chi_1(B) \in S$ et $\chi_2(B) \in \overline{S}$. Ainsi $\theta_{\mathcal{C}}(B) = 1$ et $\theta_{\overline{\mathcal{C}}}(B) = 1$. B est donc totalement caractéristique pour la classe \mathcal{C} et totalement caractéristique pour la classe $\overline{\mathcal{C}}$. Pour que la réciproque soit vraie il faut rajouter comme condition que les signes des chirotopes de \mathcal{C} soient différents des signes des chirotopes de $\overline{\mathcal{C}}$.

Nous allons maintenant voir les liens entre les taux de caractérisation et de discriminabilité.

Cas uniforme.

Propriété 6.6. Soit B une base d'un matroïde orienté uniforme. Notons θ_1 le taux de caractérisation de B pour \mathcal{C} et θ_2 le taux de caractérisation de B pour $\overline{\mathcal{C}}$. Soient S_1 et S_2 deux sous-ensembles stricts de $\{-, +\}$ tels que θ_1 soit atteint pour S_1 et θ_2 soit atteint pour S_2 . On a alors les relations suivantes

entre θ_1, θ_2 et $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$:

- Si $S_1 = S_2$ alors $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = |\theta_1 - \theta_2|$
- Si $S_1 \neq S_2$ alors $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = |\theta_1 + \theta_2 - 1|$

Preuve. Plaçons nous dans le cas où $S_1 = S_2$. Sans perte de généralité on peut supposer que $S_1 = S_2 = \{+\}$, donc $\theta_1 = \rho_{\mathcal{C}}(B, +)$ et $\theta_2 = \rho_{\overline{\mathcal{C}}}(B, +)$. Or nous avons vu que dans le cas uniforme $\rho_{\mathcal{C}}(B, -) = 1 - \rho_{\mathcal{C}}(B, +)$. Ainsi $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, +) = (\theta_1 + 1 - \theta_2 - (1 - \theta_1) - \theta_2)/2$, donc $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, +) = \theta_1 - \theta_2$. Comme $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, -) = -\tau(B, \mathcal{C}, \overline{\mathcal{C}}, +)$ dans le cas uniforme, on en déduit le résultat.

Plaçons nous dans le cas où $S_1 \neq S_2$. Sans perte de généralité on peut supposer que $S_1 = \{+\}$ et $S_2 = \{-\}$. En remplaçant θ_2 par $1 - \theta_2$ dans la preuve précédente, on obtient que $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = |\theta_1 + \theta_2 - 1|$. \square

Cas non-uniforme. Dans le cas général les valeurs de $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ de la Proposition 6.6 sont des bornes inférieures, c'est-à-dire que nous avons

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}) \geq |\theta_1 - \theta_2| \quad \text{et} \quad \tau(B, \mathcal{C}, \overline{\mathcal{C}}) \geq |\theta_1 + \theta_2 - 1|$$

Nous allons voir un exemple dans lequel ni le taux de caractérisation de \mathcal{C} , ni celui de $\overline{\mathcal{C}}$ ne servent à calculer $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$.

Soit B un $(d+1)$ -uplet de $\mathcal{B}(\mathcal{E})$ dont les valeurs de $\rho_{\mathcal{C}}(B, \{s\})$ et de $\rho_{\overline{\mathcal{C}}}(B, \{s\})$ sont données par le Tableau 6.5.

s	-	0	+
$\rho_{\mathcal{C}}(B, \{s\})$	45%	5%	50%
$\rho_{\overline{\mathcal{C}}}(B, \{s\})$	40%	30%	30%

TABLE 6.5 – Exemples de valeurs de $\rho_{\mathcal{C}}(B, \{s\})$ et $\rho_{\overline{\mathcal{C}}}(B, \{s\})$ en fonction de s

En reprenant les notations de la Proposition 6.6, on remarque dans le Tableau 6.5 que $S_1 = \{+\}$, $\theta_1 = 45\%$ et que $S_2 = \{-\}$ et $\theta_2 = 40\%$. En calculant le taux de discriminabilité on trouve $\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = 0,30 - 0,05 = 0,25$. Ce taux est atteint pour le signe 0 qui ne correspond ni à S_1 , ni à S_2 . Dans cet exemple, ni θ_1 ni θ_2 n'interviennent dans le calcul de $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$. Nous avons donc bien pour tout $(d+1)$ -uplet B

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}) \geq |\theta_1 - \theta_2| \quad \text{et} \quad \tau(B, \mathcal{C}, \overline{\mathcal{C}}) \geq |\theta_1 + \theta_2 - 1|$$

6.5 Calcul des paramètres de caractérisations faibles et fortes

Tout comme pour les méthodes de classification supervisée, l'objectif est de trouver les valeurs des paramètres afin de minimiser les erreurs de classement. Nous allons expliquer la méthode pour déterminer les valeurs des paramètres dans le cas d'une caractérisation forte, puis nous expliquerons les modifications dans cette méthode s'il s'agit de former une caractérisation faible. L'algorithme permettant de calculer ces paramètres est présenté dans la Section 8.2.

bien classé
par notre
méthode

On dit qu'un chirotope χ est bien classé par notre méthode si χ appartient à la fois à \mathcal{C} et à $\mathcal{C}_{\mathcal{D},T,q}$, ou à la fois à $\overline{\mathcal{C}}$ et à $\mathcal{C}'_{\mathcal{D},T,q}$. De même, on dit que χ est mal classé par notre méthode si χ appartient à la fois à \mathcal{C} et à $\mathcal{C}_{\mathcal{D},\overline{T},q}$, ou à la fois à $\overline{\mathcal{C}}$ et à $\mathcal{C}_{\mathcal{D},T,q}$. On dit que χ n'est pas classé par notre méthode si M n'appartient ni à $\mathcal{C}_{\mathcal{D},T,q}$ ni à $\mathcal{C}_{\mathcal{D},\overline{T},q}$.

mal classé
par notre
méthode
pas classé
par notre
méthode

En pratique l'ensemble $\mathcal{B}(\mathcal{E})$ est généralement composé d'un grand nombre d'éléments. Il ne serait donc pas possible de tester tous les sous-ensembles \mathcal{D} de $\mathcal{B}(\mathcal{E})$. Nous utilisons donc cette méthode de classification en prenant pour \mathcal{D} des ensembles de $(d+1)$ -uplets dont les taux de discriminabilité sont proches de 1. Dans ce cas le nombre de chirotopes de \mathcal{M} bien classés par notre méthode est important et nous nous intéressons donc aux chirotopes qui ne sont pas bien classés ou pas classés pour notre méthode.

Soient χ un chirotope de \mathcal{M} et $B \in \mathcal{D}$. On définit la fonction $\delta(M, B, T, \mathcal{C}, \overline{\mathcal{C}})$ de la façon suivante :

$$\delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) = \begin{cases} 1 & \text{si } \chi \in \mathcal{C} \text{ et } \chi(B) \in \overline{T(B)} \\ & \text{ou} \\ & \text{si } \chi \in \overline{\mathcal{C}} \text{ et } \chi(B) \in T(B) \\ 0 & \text{sinon} \end{cases}$$

Ainsi, cette fonction retourne 1 si et seulement si le chirotope χ est mal classé par notre méthode en prenant $\mathcal{D} = \{B\}$.

Lien avec les notions de bien et mal classé. Pour tout chirotope χ de \mathcal{M} , pour tout $(d+1)$ -uplet de \mathcal{B} et pour toute liste T de $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$, on a soit $\chi(B) \in T(B)$ soit $\chi(B) \in \overline{T(B)}$. Ainsi si $\chi \in \mathcal{C}$, on a soit $\chi(B) \in T(B)$ soit $\delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) = 1$. On en déduit que si $\chi \in \mathcal{C}$,

$$\#\mathcal{D} = \sum_{B \in \mathcal{D}} \delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) + \#\{B \in \mathcal{D} \mid \chi(B) \in T(B)\}$$

Ainsi si $\chi \in \mathcal{C}$,

$$\chi \in \mathcal{C}_{\mathcal{D},T,q} \iff \sum_{B \in \mathcal{D}} \delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) \leq \#\mathcal{D} - q$$

De même, si $\chi \in \overline{\mathcal{C}}$,

$$\chi \in \mathcal{C}_{\mathcal{D},\overline{T},q} \iff \sum_{B \in \mathcal{D}} \delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) \leq \#\mathcal{D} - q$$

On a alors :

$$\begin{aligned} \sum_{B \in \mathcal{D}} \delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) &\leq \#\mathcal{D} - q &\iff &\chi \text{ est bien classé par notre méthode} \\ \#\mathcal{D} - q < \sum_{B \in \mathcal{D}} \delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) < q &&\iff &\chi \text{ n'est pas classé par notre méthode} \\ q &\leq \sum_{B \in \mathcal{D}} \delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}}) &\iff &\chi \text{ est mal classé par notre méthode} \end{aligned}$$

Calcul des paramètres d'une caractérisation forte. Pour obtenir les paramètres d'une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$, on commence par fixer aléatoirement une liste T de $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$. On construit ensuite une matrice A telle que chaque ligne correspond à un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$, chaque colonne correspond à un chirotope χ de \mathcal{M} et que le coefficient correspondant au $(d+1)$ -uplet B et au chirotope χ (noté $A_{\chi,B}$) soit égal à $\delta(\chi, B, T, \mathcal{C}, \overline{\mathcal{C}})$. On cherche alors un sous-ensemble

de lignes \mathcal{D} tel que la somme des coefficients de ces lignes soit strictement inférieure à $\#\mathcal{D}/2$ pour chaque colonne. Si on trouve un tel sous-ensemble de lignes, en notant max le maximum des sommes des colonnes, on prend alors $q = \#\mathcal{D} - max$. On rappelle que la liste \overline{T} s'obtient en prenant pour chaque $(d+1)$ -uplet B de \mathcal{D} le complémentaire de $T(B)$ (c'est-à-dire pour tout $B \in \mathcal{D}$, $\overline{T}(B) = \{-, 0, +\} \setminus T(B)$). On a alors trouvé les paramètres \mathcal{D} , T , \overline{T} et q tels que $(\mathcal{I}_{\mathcal{D}, T, q}, \mathcal{I}_{\mathcal{D}, \overline{T}, q})$ soit une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$.

Calcul des paramètres d'une caractérisation faible. Pour le calcul des paramètres d'une caractérisation faible du couple $(\mathcal{C}, \overline{\mathcal{C}})$, on fixe également de façon aléatoire une liste T de $\mathcal{T}(\mathcal{D}, \mathcal{C}, \overline{\mathcal{C}})$. On construit ensuite deux matrices A et A' de taille respectivement $\#\mathcal{B}(\mathcal{E}) \times \mathcal{C}$ et $\#\mathcal{B}(\mathcal{E}) \times \overline{\mathcal{C}}$ de la même façon que pour une caractérisation forte. On cherche alors de la même façon, un sous-ensemble de lignes \mathcal{D} tel que la somme des coefficients de ces lignes soit strictement inférieure à $\#\mathcal{D}/2$ pour chaque colonne de A et un entier q tel que $q = \#\mathcal{D} - max$ où max est le maximum des sommes des colonnes de A . On a alors que $\mathcal{I}_{\mathcal{D}, T, q}$ est une caractérisation faible de $(\mathcal{C}, \overline{\mathcal{C}})$ si et seulement si pour toute colonne de A' la somme des coefficients des lignes de \mathcal{D} est strictement inférieure à q . Ainsi si ces valeurs pour \mathcal{D} et q ne forment pas une caractérisation faible de $(\mathcal{C}, \overline{\mathcal{C}})$, on cherche de nouveau un sous-ensemble de lignes \mathcal{D} .

Il est important de noter que cette méthode ne fait pas de différence entre deux bases dont les taux de discriminabilité sont différents. Il semblerait logique de donner plus d'importance aux bases qui ont le taux le plus élevé dans notre méthode de classification.

6.6 Exemple récapitulatif

Nous allons illustrer la méthode présentée dans ce chapitre par un exemple de sept configurations de quatre points en deux dimensions. La Table 6.6 représente chaque configuration de quatre points en deux dimensions et indique le chirotope associé à chacune d'elles. Ces sept chirotopes ont été séparés en deux sous-ensembles \mathcal{C} et $\overline{\mathcal{C}}$. On notera \mathcal{M} l'ensemble des sept chirotopes.

On remarque dans cet exemple que le triplet de points 124 est fixe puisque pour tous les chirotopes, le signe de ce triplet vaut $+$. Ainsi nous n'utiliserons pas ce triplet pour caractériser les classes.

Triplets caractéristiques. La première notion que nous avons vu est celle de $(d+1)$ -uplets caractéristiques. Pour calculer le taux de caractérisation de la classe \mathcal{C} , nous devons utiliser $\rho_{\mathcal{C}}(B, S)$ qui est la proportion de chirotopes de la classe \mathcal{C} dont le signe de B appartient à S . La Table 6.7 indique la proportion de chirotopes de \mathcal{C} , $\overline{\mathcal{C}}$ et \mathcal{M} pour le triplet 123 pour chaque sous-ensemble de signes S de $\{-, 0, +\}$.

Le taux de caractérisation est alors la plus grande valeur de chaque ligne. Ainsi, le taux de caractérisation du triplet 123 pour \mathcal{C} (noté $\theta_{\mathcal{C}}(123)$) est alors de $3/4$ et est atteint pour les ensembles de signes $\{-, 0\}$ et $\{0, +\}$. De plus, $\theta_{\overline{\mathcal{C}}}(123) = 1$ et est atteint pour les ensembles de signes $\{+\}$, $\{0, +\}$ et $\{-, +\}$, alors que $\theta_{\mathcal{M}}(123) = 6/7$ et est atteint pour $\{0, +\}$. Ainsi le triplet 123 est totalement caractéristique de $\overline{\mathcal{C}}$ pour $\{+\}$.

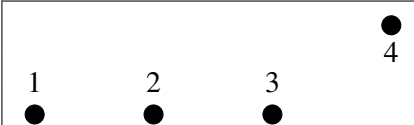
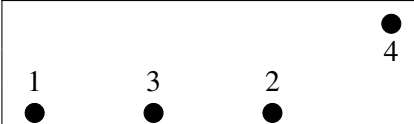
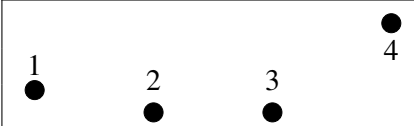
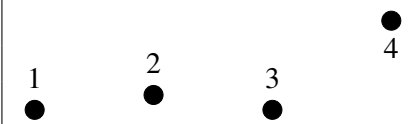
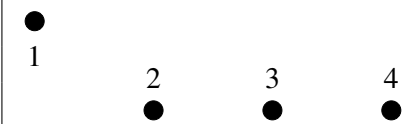
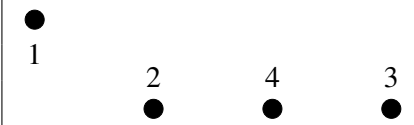
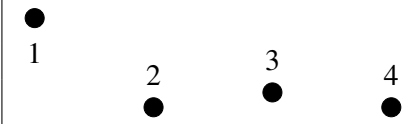
Configurations de points en 2D	Chirotopes				Classes	
	(1,2,3)	(1,2,4)	(1,3,4)	(2,3,4)		
	χ_1	0	+	+	+	\mathcal{E}
	χ_2	0	+	+	-	
	χ_3	+	+	+	+	
	χ_4	-	+	+	+	
	χ_5	+	+	+	0	$\overline{\mathcal{E}}$
	χ_6	+	+	-	0	
	χ_7	+	+	-	+	

TABLE 6.6 – Exemples de 7 configurations de 4 points en 2 dimensions et leurs chirotopes

un $(d+1)$ -uplet totalement caractéristique de \mathcal{C} et de $\overline{\mathcal{C}}$ respectivement pour $\{-, 0\}$ et $\{+\}$. Par contre le taux de caractérisation de B pour \mathcal{M} est de 87,5%. Ce taux est atteint pour $\{0, +\}$ et pour $\{-, +\}$.

S	$\{-\}$	$\{0\}$	$\{+\}$	$\{-, 0\}$	$\{0, +\}$	$\{-, +\}$
$\rho_{\mathcal{C}}(123, S)$	1/4	1/2	1/4	3/4	3/4	1/2
$\rho_{\overline{\mathcal{C}}}(123, S)$	0	0	1	0	1	1
$\rho_{\mathcal{M}}(123, S)$	1/7	2/7	4/7	3/7	6/7	5/7

TABLE 6.7 – Valeurs de $\rho_{\mathcal{C}}(123, S)$, $\rho_{\overline{\mathcal{C}}}(123, S)$ et $\rho_{\mathcal{M}}(123, S)$ en fonction de l'ensemble de signes S

Triplets discriminants. On rappelle que le taux de discriminabilité d'un triplet B entre \mathcal{C} et $\overline{\mathcal{C}}$ (noté $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$) est défini de la façon suivante :

$$\tau(B, \mathcal{C}, \overline{\mathcal{C}}) = \max_{\substack{S \subset \{-, 0, +\} \\ S \neq \{-, 0, +\}}} \tau(B, \mathcal{C}, \overline{\mathcal{C}}, S)$$

où $\tau(B, \mathcal{C}, \overline{\mathcal{C}}, S) = \rho_{\mathcal{C}}(B, S) - \rho_{\overline{\mathcal{C}}}(B, S)$

Ainsi le taux de discriminabilité d'un triplet 123 entre \mathcal{C} et $\overline{\mathcal{C}}$ est la plus grande valeur de la Table 6.8. On a alors $\tau(123, \mathcal{C}, \overline{\mathcal{C}}) = 3/4$ et est atteint par l'ensemble de signes $\{-, 0\}$. En procédant de la même façon, on obtient $\tau(124, \mathcal{C}, \overline{\mathcal{C}}) = 0$, $\tau(134, \mathcal{C}, \overline{\mathcal{C}}) = 2/3$ atteint entre autre pour $\{+\}$ et $\tau(234, \mathcal{C}, \overline{\mathcal{C}}) = 2/3$ atteint pour $\{+, -\}$.

S	$\{-\}$	$\{0\}$	$\{+\}$	$\{-, 0\}$	$\{0, +\}$	$\{-, +\}$
$\tau(123, \mathcal{C}, \overline{\mathcal{C}}, S)$	1/4	1/2	-3/4	3/4	-1/4	-1/2

TABLE 6.8 – Valeurs de $\tau(123, \mathcal{C}, \overline{\mathcal{C}}, S)$ en fonction de S

Caractérisation forte. On rappelle qu'une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$ dépend de quatre paramètres : un ensemble \mathcal{D} de triplets, deux listes T et \overline{T} de taille $\#\mathcal{D}$ contenant des sous-ensembles stricts de $\{-, 0, +\}$ et un entier q tel que $\#\mathcal{D}/2 < q \leq \#\mathcal{D}$. Si ces paramètres forment une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$, cela signifie que $\mathcal{C} = \mathcal{I}_{\mathcal{D}, T, q}$ et $\overline{\mathcal{C}} = \mathcal{I}_{\mathcal{D}, \overline{T}, q}$, où $\mathcal{I}_{\mathcal{D}, T, q}$ est l'ensemble de chirotopes défini de la façon suivante :

$$\chi \in \mathcal{I}_{\mathcal{D}, T, q} \iff \#\{B \in \mathcal{D} \mid \chi(B) \in T(B)\} \geq q$$

Pour calculer une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$, on commence par trier les triplets selon le taux de discriminabilité entre \mathcal{C} et $\overline{\mathcal{C}}$. On note \mathcal{B} l'ensemble ordonné des triplets. On a alors $B = (123, 134, 234, 124)$.

D'après ce que nous avons vu dans la Section 6.2.2, on peut prendre pour T une liste contenant des sous-ensembles stricts de $\{-, 0, +\}$ minimum pour l'inclusion tel que pour tout triplet B de \mathcal{B} $\tau(B, \mathcal{C}, \overline{\mathcal{C}})$ soit atteint pour $T(B)$. Ainsi dans cet exemple on prendra $T = (\{-, 0\}, \{+\}, \{+, -\}, \{-\})$. Lorsque nous avons défini la caractérisation forte dans le cas non-uniforme, nous avons vu que la liste \overline{T} s'obtient en prenant pour chaque $(d+1)$ -uplet B de \mathcal{D} le complémentaire de $T(B)$ (c'est-à-dire pour tout $B \in \mathcal{D}$, $\overline{T}(B) = \{-, 0, +\} \setminus T(B)$). On a alors $\overline{T} = (\{+\}, \{-, 0\}, \{0\}, \{0, +\})$.

Comme aucun triplet n'a un taux de discriminabilité entre \mathcal{C} et $\overline{\mathcal{C}}$ égal à 1, il n'est pas possible d'avoir une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$ avec un ensemble de triplet \mathcal{D} de taille 1. En effet, si l'on prend par exemple $\mathcal{D} = (123)$, on a alors $q = 1$. On a donc $\mathcal{I}_{\mathcal{D}, T, q} = \{\chi \in \mathcal{M} \mid \chi(123) \in \{-, 0\}\}$ et $\mathcal{I}_{\mathcal{D}, \overline{T}, q} = \{\chi \in \mathcal{M} \mid \chi(123) \in \{+\}\}$. Ainsi, $\mathcal{I}_{\mathcal{D}, T, q} = \{\chi_1, \chi_2, \chi_4\}$ et $\mathcal{I}_{\mathcal{D}, \overline{T}, q} = \{\chi_3, \chi_5, \chi_6, \chi_7\}$. χ_3 est alors mal classé par notre méthode et $(\mathcal{I}_{\mathcal{D}, T, q}, \mathcal{I}_{\mathcal{D}, \overline{T}, q})$ n'est pas une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$.

Nous allons essayer d'obtenir une caractérisation forte avec $\mathcal{D} = (123, 134)$. On a alors $q = 2$ (car $q > \#\mathcal{D}/2$). Ainsi on a $\chi \in \mathcal{I}_{\mathcal{D}, T, q}$ si et seulement si $\chi(123) \in \{-, 0\}$ et $\chi(134) \in \{+\}$. On a donc $\mathcal{I}_{\mathcal{D}, T, q} = \{\chi_1, \chi_2, \chi_4\}$. De même, $\chi \in \mathcal{I}_{\mathcal{D}, \overline{T}, q}$ si et seulement si $\chi(123) \in \{+\}$ et $\chi(134) \in \{-, 0\}$. On a donc $\mathcal{I}_{\mathcal{D}, \overline{T}, q} = \{\chi_6, \chi_7\}$. Ainsi χ_3 et χ_5 ne sont pas classés par notre méthode et $(\mathcal{I}_{\mathcal{D}, T, q}, \mathcal{I}_{\mathcal{D}, \overline{T}, q})$ n'est pas une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$.

Les autres sous-ensembles de \mathcal{B} de taille 2 ne nous fournissent pas de caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$. Par contre on obtient une caractérisation forte de $(\mathcal{C}, \overline{\mathcal{C}})$ en prenant $\mathcal{D} = (123, 134, 234)$ et $q = 2$.



Partie IV. Implémentation logicielle

Dans cette partie nous présentons les problèmes théoriques et pratiques que nous avons résolus pour écrire les algorithmes permettant de calculer notre méthode. Les programmes correspondants ont été réalisés en C++ et les codes sont disponibles à l'adresse http://www2.lirmm.fr/~sol/these_codes.

Le Chapitre 7 est consacré au calcul du chirotope. D'abord, nous présentons l'algorithme permettant de calculer le chirotope d'une configuration de points en dimension quelconque. Nous en profitons pour décrire aussi les algorithmes de calcul des circuits et cocircuits d'une configuration de points en dimension quelconque. Nous expliquons ensuite la méthode employée pour garantir l'exactitude des calculs et nous réalisons une étude de la complexité des algorithmes en indiquant les temps de calcul en fonction du nombre de points. Comme le chirotope d'une configuration de points est un vecteur de signes où chaque signe correspond à un $(d+1)$ -uplet de points, nous présentons dans la Section 7.2, comment calculer l'indice d'un $(d+1)$ -uplet de points dans ce vecteur. La dernière section de ce chapitre (Section 7.3) présente le choix qui ont été fait concernant le format des fichiers de données et des fichiers retournés par le programme.

Le Chapitre 8 est consacré aux problèmes théoriques et pratiques qui se sont posés lors de la programmation des autres étapes de notre méthode. D'abord, nous présentons l'algorithme calculant l'ensemble des bases fixes issues d'ordres sur les coordonnées des points. Puis nous présentons un algorithme qui calcule les paramètres nous permettant d'avoir une caractérisation forte (au sens de la définition de la Section 6.2) de classes de chirotopes. Enfin nous étudions la symétrie (et l'asymétrie) d'une configuration de points de \mathbb{R}^d et d'un hyperplan de \mathbb{R}^d . On s'intéresse à deux demi-structures, la première est formée des points appartenant à l'hyperplan et des points qui sont d'un côté de l'hyperplan et la deuxième est formée des points appartenant à l'hyperplan et des points qui sont de l'autre côté de l'hyperplan. En particulier, nous présentons un programme permettant à partir du chirotope de la configuration de points, d'extraire le chirotope d'une demi-structure et de calculer le chirotope du symétrique de l'autre demi-structure par rapport à l'hyperplan.

Implémentation du calcul du chirotope

7.1 Matroïde orienté associé à une configuration de points en dimension quelconque

Pour pouvoir comparer les structures anatomiques, il nous faut d'abord calculer le matroïdes orientés associé à chaque échantillon. Un programme informatique du nom de TOPCOM disponible librement sur le site <http://www.rambau.wm.uni-bayreuth.de/TOPCOM/> permet de calculer le matroïde orienté associé à un ensemble de points. Ce programme a été réalisé avec pour objectif de calculer les triangulations d'un ensemble de points en dimension quelconque [58]. Nous aurions pu utiliser ce programme pour calculer les matroïdes orientés mais nous avons choisi de créer notre propre programme afin notamment de pouvoir modifier plus facilement le programme dans le but entre autre de garantir l'exactitude des calculs.

Nous avons donc mis en place dans le cadre de cette thèse un programme en C++ qui calcule les bases, les cocircuits ainsi que les circuits d'un ensemble de points affine (noté \mathcal{E}) de dimension quelconque (noté d). On suppose que le nombre de points (noté n) est strictement plus grand que la dimension de l'espace auxquels ils appartiennent. Si cette condition n'est pas respectée alors l'ensemble des circuits est vide et tous les chirotopes sont nuls.

7.1.1 Méthodes de calcul du chirotope, des cocircuits et des circuits

Nous avons vu dans le Chapitre 2 que l'on pouvait caractériser un matroïde orienté par son chirotope, par l'ensemble de ses cocircuits ou par l'ensemble de ses circuits. Les calculs sont réalisés à partir de l'ensemble des vecteurs (notés \mathcal{V}) associés aux points. La première étape du programme est donc d'obtenir la matrice des coordonnées des vecteurs de \mathcal{V} en ajoutant une ligne de 1 à cette matrice pour obtenir celle des coordonnées des vecteurs. La matrice des coordonnées des vecteurs de \mathcal{V} est donc de dimension $(d + 1) \times n$.

Calcul du chirotope

Algorithme 7.1 : Calcul des chirotopes

Entrée : La matrice des coordonnées des vecteurs de \mathcal{V}

Sortie : Le chirotope χ écrit sous forme d'un vecteur

Initialiser χ à un vecteur nulle de taille $\binom{n}{d+1}$;

Initialiser k à 0;

pour tous les $(d+1)$ -uplets $\{v_{i_1}, \dots, v_{i_{d+1}}\}$ de vecteurs de \mathcal{V} **faire**

 Ecrire la matrice $Mat = (v_{i_1} \ \dots \ v_{i_{d+1}})$;

 Calculer $det(Mat)$;

$\chi_k = det(Mat)$;

$k = k + 1$;

Le chirotope d'un matroïde orienté associé à une configuration de points est un vecteur de signes appartenant à $\{-, 0, +\}$ de taille $\binom{n}{d+1}$. Chaque signe correspond à un $(d+1)$ -uplet de \mathcal{V} et est égal au signe du déterminant d'une matrice $(d+1) \times (d+1)$ dont les colonnes sont les coordonnées des $d+1$ vecteurs.

Pour calculer le chirotope, on utilise l'Algorithme 7.1 en prenant tous les $(d+1)$ -uplets de \mathcal{V} dans l'ordre lexicographique.

Calcul des cocircuits

Algorithme 7.2 : Calcul des cocircuits

Entrée : La matrice des coordonnées des vecteurs de \mathcal{V}

Sortie : La liste l de tous les cocircuits

Initialiser l à la liste vide;

pour tous les d -uplets $\{v_{i_1}, \dots, v_{i_d}\}$ de vecteurs de \mathcal{V} **faire**

 Ecrire la matrice $M = (v_{i_1} \ \dots \ v_{i_d})$;

 Calculer $rang(M)$;

si le rang est égal à d **alors**

 Initialiser C^* à la liste vide;

pour tous les vecteurs v_j de \mathcal{V} **faire**

 Ecrire la matrice $Mat = (v_j \ v_{i_1} \ \dots \ v_{i_d})$;

 Calculer $det(Mat)$;

 Ajouter le signe du déterminant à la liste C^* ;

 Ajouter C^* à la liste l ;

Le calcul des cocircuits d'un matroïde orienté est un peu plus délicat. Nous allons donc détailler l'Algorithme 7.2. Pour calculer un cocircuit, on doit trouver un hyperplan (c'est-à-dire d vecteurs de \mathcal{V} qui sont linéairement indépendants) avant de regarder dans quel demi-espace se trouve chaque vecteur de \mathcal{V} . Pour savoir si d vecteurs forment un hyperplan, une méthode est d'écrire la matrice dans laquelle chaque colonne correspond à un des d vecteurs (la matrice est donc constituée de $d+$

1 lignes et de d colonnes) et de calculer son rang. Si le rang de la matrice est égal à d , les d vecteurs forment bien un hyperplan. On va donc prendre tous les sous-ensembles de \mathcal{V} à d éléments et tester si chaque sous-ensemble constitue une matrice de rang égal à d . Ensuite si un sous-ensemble forme un hyperplan H , pour chaque vecteur v_j de \mathcal{V} on crée une matrice dont la première colonne est composée des coordonnées de v_j et les d autres colonnes sont composées des coordonnées des vecteurs formant l'hyperplan H . On calcule alors le signe du déterminant de cette matrice. Comme le signe du déterminant d'une matrice indique l'orientation des vecteurs par rapport à la base sur laquelle est écrite la matrice; si les déterminants de deux matrices ont le même signe, alors les vecteurs ont la même orientation. Ainsi si le déterminant d'une telle matrice est nul, alors le vecteur composant sa première colonne appartient à l'hyperplan H et si les déterminants de deux matrices ont le même signe, cela nous indique que les vecteurs composant la première colonne de chaque matrice sont dans le même demi-espace délimité par H .

Calcul des circuits

Comme nous l'avons vu dans le Chapitre 2, les circuits d'un matroïde orienté sont les cocircuits du matroïde orienté dual. Pour calculer les circuits du matroïde nous allons donc calculer le matroïde orienté dual puis nous allons utiliser l'Algorithme 7.2 à partir de ce matroïde orienté dual.

Algorithme 7.3 : Calcul du matroïde orienté dual

Entrée : La matrice M des coordonnées des vecteurs de \mathcal{V}

Sortie : La matrice M^* correspondant au matroïde orienté dual

Prendre $d + 1$ premiers vecteurs $\{v_{i_1}, \dots, v_{i_{d+1}}\}$ de \mathcal{V} ;

tant que $\det(v_{i_1} \dots v_{i_{d+1}}) = 0$ **faire**

 └ Prendre un autre $(d + 1)$ -uplet de vecteurs;

 Ecrire la matrice M en commençant par ces $d + 1$ vecteurs;

 Ecrire M sous forme échelonnée réduite (c'est-à-dire sous la forme $(I_{d+1}|A)$);

 Ecrire la matrice $M^* = (-{}^t A | I_{n-(d+1)})$;

Pour calculer le matroïde orienté dual, on utilise l'Algorithme 7.3. On note M la matrice des coordonnées des vecteurs de \mathcal{V} (c'est-à-dire que M est la matrice de taille $(d + 1) \times n$ dont la i -ème colonne est composée des coordonnées du i -ème vecteur de \mathcal{V}). Cet algorithme consiste à écrire M sous la forme $(I_{d+1}|A)$ (où I_{d+1} est la matrice identité de taille $(d + 1) \times (d + 1)$ et A est une matrice de taille $(d + 1) \times (n - (d + 1))$) puis à écrire la matrice $M^* = (-{}^t A | I_{n-(d+1)})$. Le matroïde orienté dual se calcule alors à partir de la matrice M^* . L'Algorithme 7.3 se déroule donc en deux étapes.

Pour la première étape, il faut commencer par trouver $d + 1$ vecteurs parmi les n vecteurs de \mathcal{V} de sorte que ces $d + 1$ vecteurs forment une base de \mathbb{R}^{d+1} . On commence par tester si les vecteurs $\{v_1, \dots, v_{d+1}\}$ forment une base de \mathbb{R}^{d+1} ce qui revient à regarder si $\det(v_1, \dots, v_{d+1}) \neq 0$. Si ce déterminant est nul, les vecteurs $\{v_1, \dots, v_{d+1}\}$ ne forment pas une base de \mathbb{R}^{d+1} et il faut donc changer au moins l'un de ces vecteurs. Une fois que l'on a les $d + 1$ vecteurs formant une base on réécrit la matrice (v_1, \dots, v_n) en plaçant en premier ces $d + 1$ vecteurs. La deuxième étape consiste alors à écrire la matrice sous la forme échelonnée réduite (c'est-à-dire sous la forme $(I_{d+1}|A)$). Pour écrire la matrice sous cette forme, nous utilisons la méthode du pivot de Gauss. Il ne nous reste plus qu'à calculer ${}^t A$ pour avoir la matrice $(-{}^t A | I_{n-(d+1)})$ à partir de laquelle nous pouvons calculer le

matroïde orienté dual. Il nous suffit alors de calculer les cocircuits à partir de cette matrice pour obtenir les circuits du matroïde orienté associé à l'ensemble de points.

Calcul du déterminant et du rang

Pour réaliser les calculs des bases, cocircuits et circuits du matroïde orienté associé à l'ensemble de points \mathcal{E} , on utilise des calculs de déterminants et de rangs de matrices. Ces deux calculs peuvent notamment être réalisés grâce à la méthode du pivot de Gauss qui est utilisée dans le calcul du matroïde dual. Ainsi nous avons décidé pour des questions de simplicité de programmation de réaliser ces calculs par la méthode du pivot de Gauss, bien qu'il existe des méthodes plus performantes [35]. Il est donc envisageable de changer de méthode pour les calculs du déterminant et du rang. Pour plus de renseignements sur la méthode du pivot de Gauss ou sur les matrices, je recommande le Livre [39].

7.1.2 Exactitude des calculs

Dès le début, l'exactitude des calculs s'est avérée comme un problème critique. En effet, le calcul des matroïdes orientés est très sensible aux perturbations lorsque le déterminant est proche de zéro. Lors du calcul d'un cocircuit, par exemple, le signe d'un point indique si le point est d'un coté ou de l'autre de l'hyperplan ou si il appartient à cet hyperplan. Si le calcul du cocircuit n'est pas exact on pourrait se retrouver avec des points qui sont proches de l'hyperplan pour lesquels le calcul indiquerait que le point appartient à l'hyperplan voire même qu'il est dans l'autre demi-espace.

Pour résoudre ce problème, nous avons décidé de représenter les valeurs des coordonnées par des fractions. Comme la méthode du pivot de Gauss n'utilise que les opérations de base (addition, soustraction, multiplication et division), nous resterons toujours dans l'espace des fractions ce qui garanti l'exactitude des calculs.

L'utilisation de fractions à la place de nombres décimaux garantie l'exactitude des calculs mais augmente le nombre de calculs à faire (mise sous le même dénominateur, simplification des fractions ...). De plus, les entiers constituant les fractions peuvent être très grands. Il a fallu trouver une méthode qui garantissait de pouvoir travailler avec des fractions de taille arbitraire tout en ayant des temps de calculs raisonnables. L'arithmétique multiprécision permet de résoudre ces problèmes et le choix s'est porté vers la bibliothèque "GNU Multiple Precision" (disponible sur le site <http://gmp.org/>). Bien entendu, la taille des fractions augmente avec le nombre de chiffres significatifs pris pour les coordonnées des points. Le temps de calcul de la méthode dépendra donc fortement de ce nombre de chiffres significatifs.

7.1.3 Complexité théorique et temps de calcul

Comme nous l'avons vu précédemment, les méthodes de calculs des chirotopes, des cocircuits ainsi que des circuits sont basées sur la méthode du pivot de Gauss. Cette méthode se passe en 2 temps, la matrice est tout d'abord mise sous forme triangulaire supérieure, puis sous sa forme échelonnée réduite.

Pour le calcul des déterminants et des rangs des matrices, seule la première étape est nécessaire. En notant l le nombre de lignes de la matrice, c son nombre de colonnes et m le minimum des 2, la complexité de cette étape est alors en $O(m \times l \times c)$. Les calculs de rang et de déterminant lors de la

recherche des chirotopes et des cocircuits se font donc en $O(d^3)$. La deuxième étape de la méthode du pivot de Gauss est quand à elle en $O(l^2 \times c)$.

Pour le calcul des chirotopes, il faut faire un calcul de déterminant pour tous les $(d + 1)$ -uplets de vecteurs de \mathcal{V} . Donc l'Algorithme 7.1 de calcul des chirotopes est en $O(d^3 \times n^{d+2})$.

En ce qui concerne le calcul des cocircuits, il faut faire un calcul de rang d'une matrice de taille $(d + 1) \times d$ puis pour tous les vecteurs de \mathcal{V} il faut calculer le déterminant d'une matrice de taille $(d + 1) \times (d + 1)$ et ce pour tous les d -uplets de vecteurs de \mathcal{V} . Donc l'Algorithme 7.2 de calcul des cocircuits est en $O((d^3 + n \times d^3) \times n^{d+1}) = O(n^{d+2} \times d^3)$.

Le calcul des circuits est un peu plus délicat. Il faut commencer par calculer le matroïde orienté dual. Dans le pire des cas, pour trouver une base de \mathbb{R}^{d+1} il faut prendre tous les $(d + 1)$ -uplets de vecteurs de \mathcal{V} (ce qui est très rare en pratique puisque les points sont souvent en position générale). Puis on utilise les 2 étapes de la méthode du pivot de Gauss. L'Algorithme 7.3 de calcul du matroïde orienté dual est donc en $O(n^{d+2})$ dans le pire des cas mais si les points sont en position générale, cet algorithme est en $O(d^2 \times n)$. Ensuite pour obtenir les circuits, on utilise le calcul des cocircuits avec cette nouvelle matrice de dimension $n - (d + 1) \times n$. Cela revient à calculer le rang d'une matrice taille $(n - (d + 1)) \times (n - (d + 1) - 1)$ puis pour tous les vecteurs de \mathcal{V} il faut calculer le déterminant d'une matrice de taille $(n - (d + 1)) \times (n - (d + 1))$ et ce pour tous les $(n - (d + 1) - 1)$ -uplets de vecteurs de \mathcal{V} (c'est-à-dire pour tous les $(n - d - 2)$ -uplets). Or $\binom{n}{n-d-2} = \binom{n}{d+2}$, donc l'algorithme de calcul des circuits est en $O(((n - d)^3 + n \times (n - d)^3) \times n^{d+3}) \approx O(n^{d+7})$.

Il faut noter que le programme a pour but de comparer des formes anatomiques 3D. On a donc $d = 3$ ce qui revient à dire que les algorithmes de calculs de chirotopes et de cocircuits sont polynomiaux en n (ils sont en $O(n^5)$). Par contre le calcul des circuits s'effectue en $O(n^{10})$. Le fait que $\binom{n}{d+2}$ soit très inférieur à n^{d+3} , ajouté à quelques optimisations pour diminuer le nombre de calculs font que les temps de calculs restent en pratique assez faibles. Par exemple, pour un ensemble de 16 points en trois dimensions dont les coordonnées ont six chiffres significatifs, le calcul de l'ensemble des chirotopes ainsi que celui de l'ensemble des cocircuits s'effectuent en environ une seconde chacun et le temps de calcul de l'ensemble des circuits est de l'ordre de 50 secondes sur un ordinateur standard.

7.2 Calcul de l'indice d'un $(d + 1)$ -uplet

Lorsque nous étudions et comparons des chirotopes, nous sommes amenés à nous focaliser sur des sous-ensembles de landmarks ou sur certains $(d + 1)$ -uplets. Or les $(d + 1)$ -uplets sont triés selon l'ordre lexicographique. Nous avons alors cherché un moyen de déterminer l'indice d'un $(d + 1)$ -uplet dans cet ordre.

En C++, les indices des cases d'un tableau de taille k sont numérotés de 0 à $k - 1$. Nous allons donc chercher l'indice d'un $(d + 1)$ -uplet de $\{0, \dots, n - 1\}$ dans l'ordre lexicographique avec comme convention que le premier $(d + 1)$ -uplet dans cet ordre à pour indice 0 et que le dernier $(d + 1)$ -uplet à pour indice $\binom{n}{d+1} - 1$.

Le calcul de l'indice d'un $(d + 1)$ -uplet B dans l'ordre lexicographique consiste à compter le nombre de $(d + 1)$ -uplets qui précèdent B dans cet ordre. On note $B = (b_0, b_1, \dots, b_d)$ avec pour tout $i \in \{0, \dots, d\}$ $b_i \in \{0, \dots, n - 1\}$ et pour tout $(i, j) \in \{0, \dots, d\}^2$ tels que $i < j$ on ait $b_i < b_j$.

Pour calculer l'indice de B , on compte dans un premier temps le nombre de $(d+1)$ -uplets qui précèdent B dont le premier élément est strictement inférieur à b_0 . Or le nombre de $(d+1)$ -uplets dont le premier élément est 0 est égal au nombre de combinaisons de d éléments pris parmi $n-1$ éléments. Plus généralement, le nombre de $(d+1)$ -uplets dont le premier élément est k est égal au nombre de combinaisons de $(d+1-1)$ éléments pris parmi $n-(k+1)$ éléments. Ainsi en notant $n_0(B)$ le nombre de $(d+1)$ -uplets qui précèdent B dont le premier élément est strictement inférieur à b_0 , on a :

$$n_0(B) = \sum_{k=0}^{b_0} \binom{n-(k+1)}{d}$$

Maintenant, pour tout $i \in \{1, \dots, d\}$, il faut compter le nombre de $(d+1)$ -uplets qui précèdent B tels que les i premiers éléments sont les mêmes que ceux de B . En d'autres termes, on cherche le nombre de $(d+1)$ -uplets B' qui précèdent B tels que $B' = (b_0, \dots, b_{i-1}, b'_i, \dots, b'_d)$ avec $b'_i < b_i$. Soit j un entier tel que $b_{i-1} < j < b_i$. Pour les mêmes raisons que précédemment, le nombre de $(d+1)$ -uplets qui précèdent B tels que les i premiers éléments sont les mêmes que ceux de B et que le $(i+1)$ -ème élément vaut k est égal au nombre de combinaisons de $(d+1-(i+1))$ éléments pris parmi $n-(k+1)$ éléments. Ainsi en notant $n_i(B)$ le nombre de $(d+1)$ -uplets qui précèdent B tels que les i premiers éléments sont les mêmes que ceux de B , on a :

$$n_i(B) = \sum_{k=b_{i-1}+1}^{b_i-1} \binom{n-(k+1)}{d-i}$$

L'indice de B (noté $ind(B)$) se calcule donc par la formule suivante :

$$ind(B) = \sum_{k=0}^{b_0} \binom{n-(k+1)}{d} + \sum_{i=1}^n \sum_{k=b_{i-1}+1}^{b_i-1} \binom{n-(k+1)}{d-i}$$

7.3 Réflexions sur les fichiers de données et de résultats

Les fichiers de données. Lorsque nous avons commencé à programmer le calcul de chirotopes, nous avons choisi d'utiliser notre propre format. Les fichiers n'étaient alors composés presque exclusivement que des coordonnées des points. Ce format était simple à utiliser aussi bien pour la création des fichiers que pour leur lecture par un programme informatique. Cependant, nous nous sommes aperçus lors de l'utilisation pratique que des informations importantes n'étaient pas présentes dans ces fichiers, telles que les identifiants des différents échantillons ou les compositions des classes des experts. Nous avons alors décidé d'utiliser un format employé dans la communauté des morphométriciens qui est celui de Morphologika. Ce format est à la fois facile d'utilisation et contient toutes les informations dont nous avons besoin.

Les fichiers de résultats. Dans un premier temps, nous avons choisi d'utiliser un fichier de résultat par échantillon où chaque ligne correspondait à un $(d+1)$ -uplet de points. Sur chaque ligne nous

indiquions les indices des $d+1$ points suivi du signe du $(d+1)$ -uplet. Chaque fichier était donc composé de $\binom{n}{d+1}$ lignes. Ce format était pratique pour tester les programmes sur des jeux de données composés de peu d'échantillons constitués de peu de points.

Comme nos jeux de données réels étaient composés de centaines d'échantillons, nous avons alors choisi de rassembler les chirotopes de tous les échantillons dans un seul fichier. Chaque ligne correspond à un échantillon et contient le chirotope de l'échantillon sous forme d'un vecteur. Ainsi chaque ligne est composée de $\binom{n}{d+1}$ signes séparés par des espaces pour rendre la lecture du fichier plus simple. En particulier, ce format est pratique pour déterminer visuellement si un $(d+1)$ -uplet est fixe puisque dans ce cas là, tous les signes de la colonne correspondant à ce $(d+1)$ -uplet sont égaux.

Par contre, notre dernier jeu de données est composé de 60 échantillons avec 133 landmarks par échantillon. Le fichier est alors composée de 60 lignes chacune contenant $\binom{133}{4} = 12\,457\,445$ signes. Ainsi pour ces données, le fichier résultat contenant les chirotopes est composé de $60 \cdot \binom{133}{4}$ signes et pèse environ 1,4 Go. En plus du problème de stockage du fichier, la lecture d'un fichier aussi lourd par un programme informatique demande du temps et de la place en mémoire. Pour diminuer la taille du fichier on aurait pu trouver un codage permettant de diminuer le nombre de caractères à utiliser, mais un tel codage n'aurait pas permis de diminuer à la fois le temps pour la lecture du fichier et l'espace occupée sur la mémoire vive. Or nous avons remarqué qu'en pratique, la majorité des $(d+1)$ -uplets sont fixes. Nous avons alors décidé de stocker l'ensemble des chirotopes en utilisant trois fichiers. Le premier fichier indique si un $(d+1)$ -uplet est fixe ou non en stockant une valeur 0 ou 1 pour chaque $(d+1)$ -uplet. Le deuxième fichier permet de stocker les signes des $(d+1)$ -uplets fixes. Comme un $(d+1)$ -uplet est fixe si le signe de cet uplet est le même pour chaque échantillon, il suffit de stocker un seul signe par $(d+1)$ -uplet fixe. Le troisième fichier sert à stocker les $(d+1)$ -uplets non-fixes et est le seul à contenir autant de lignes que d'échantillons. Avec ce système, nous obtenons un premier fichier qui pèse environ 24 Mo (il y a 9 781 446 $(d+1)$ -uplets fixes), un deuxième fichier qui pèse environ 19 Mo et un troisième fichier ne pesant plus qu'environ 306 Mo. Ainsi on diminue par près de cinq la place occupée en mémoire. De plus, les calculs des programmes sont bien plus rapides puisqu'ils ne se font que sur les $(d+1)$ -uplets non-fixes et n'utilise en fait que le troisième fichier.

Il est important de noter que le calcul des chirotopes n'est pas ralenti par ce système de stockage en trois fichiers puisqu'il est possible de savoir si un $(d+1)$ -uplet est fixe ou non pendant le calcul des chirotopes. En effet, lors du calcul du chirotope du premier échantillon, on stocke en mémoire le chirotope du premier échantillon et une valeur binaire égale à "Vrai" pour chaque $(d+1)$ -uplet. Ensuite pour chaque autre échantillon, après le calcul du signe d'un $(d+1)$ -uplet, on compare ce signe avec celui du premier échantillon. Si les signes sont différents, on affecte la valeur "Faux" à la variable binaire correspondant à ce $(d+1)$ -uplet. A la fin des calculs des chirotopes, ces variables binaires nous permettent de remplir le premier des trois fichiers et nous permettent de déterminer simplement dans quel fichier doivent être stockés les signes.

Implémentation du reste de la méthode

8.1 Calcul de l'ensemble des bases fixes issues d'ordres sur les coordonnées des points

Dans cette section, nous allons présenter le programme permettant de calculer l'ensemble des bases fixes issues d'ordres sur les coordonnées des points.

Dans le Chapitre 3, nous avons trouvé une caractérisation (Théorèmes 3.10 et 3.11) des configurations d'ordres fixes de trois ordres totaux. Nous avons également prouvé dans la Proposition 3.3 qu'en étudiant les extensions linéaires d'une configuration d'ordres, nous pouvions déterminer si la configuration d'ordres est fixe ou non.

Nous allons reprendre l'exemple que nous avons utilisé dans la Section 3.1.2. On considère un jeu de données composé de dix landmarks anatomiques 3D relevés par des experts sur 305 crânes (voir Figure 8.1). Ces points sont pris dans un repère où l'axe \bar{x} va de la droite du crâne vers la gauche, l'axe \bar{y} va de la base du crâne vers le sommet du crâne et l'axe \bar{z} va de l'avant du crâne vers l'arrière du crâne. Comme ces landmarks proviennent de crânes, nous pouvons définir des relations anatomiques qui vont se modéliser par des relations d'ordre sur les coordonnées des landmarks.

Les configurations d'ordres sont représentés sur les Figures 8.2 et 8.3. En prenant par exemple $\mathcal{E} = \{1, 5, 9, 10\}$, nous obtenons alors la configuration d'ordres suivante.

$$\begin{aligned} 9 <_x 1 <_x 10 \quad \text{et} \quad 9 <_x 5 <_x 10 \\ 5 <_y 9 <_y 1 \quad \text{et} \quad 5 <_y 10 <_y 1 \\ 1 <_z 9 \quad \text{et} \quad 1 <_z 10 \quad \text{et} \quad 5 <_z 9 \quad \text{et} \quad 5 <_z 10 \end{aligned}$$

Nous avons vu dans la Section 3.1.2 que cette configuration d'ordres est fixe, ce qui signifie que pour ce jeu de données le quadruplet de landmarks $\{1, 5, 9, 10\}$ forme une base fixe.

Notre objectif est alors d'écrire un programme qui à partir de trois relations ordres $<_x, <_y$ et $<_z$ sur $\{1, \dots, n\}$, permet de calculer toutes les configurations fixes de trois ordres sur des sous-ensembles de taille quatre de $\{1, \dots, n\}$ qui respectent ces relations.

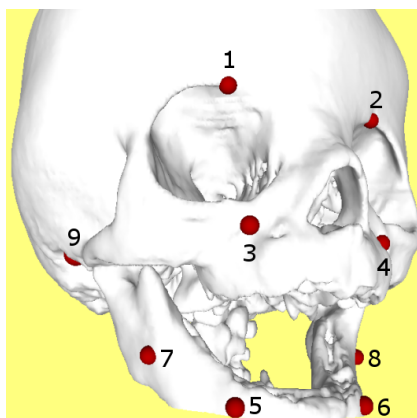


FIGURE 8.1 – Position des 10 landmarks

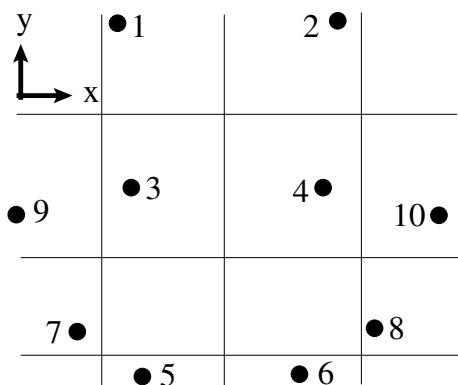


FIGURE 8.2 – Représentation des ordres sur une vue de face

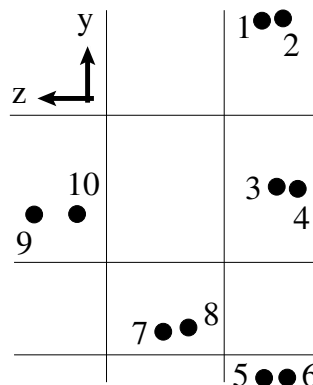


FIGURE 8.3 – Représentation des ordres sur une vue de profil

Nous avons choisi de représenter chaque relation d'ordre par une matrice de taille $n \times n$. Nous noterons respectivement R_x , R_y et R_z les matrices des relations $<_x$, $<_y$ et $<_z$ sur $\{1, \dots, n\}$. On code la relation $<_x$ de la façon suivante : pour tout $(i, j) \in \{1, \dots, n\}^2$ si $i <_x j$ alors $R_x(i, j)$ est égal à 1 et si $j <_x i$ alors $R_x(i, j)$ est égal à -1. Si par contre i et j ne sont pas comparables par relation d'ordre $<_x$ et que $i \neq j$, alors $R_x(i, j)$ est égal à 0. On procède de la même façon pour construire les deux autres matrices R_y et R_z . Pour chaque relation d'ordre, la matrice est donc symétrique. Ainsi, la relation d'ordre est codée entièrement sur la partie triangulaire inférieure (ou triangulaire supérieure) de la matrice.

Pour tout sous-ensemble \mathcal{E} de de taille quatre de $\{1, \dots, n\}$, le premier ordre de la configuration \mathcal{C} sur \mathcal{E} est codé par la sous-matrice carrée $R_x^{\mathcal{C}}$ de R_x de taille 4×4 dont les lignes et les colonnes correspondent aux éléments de \mathcal{E} . En procédant de la même façon pour les relations $<_y$ et $<_z$, on obtient un codage de chaque configuration de trois ordres sur des sous-ensembles de taille quatre de $\{1, \dots, n\}$ qui respectent les relations $<_x$, $<_y$ et $<_z$ par trois matrices $R_x^{\mathcal{C}}$, $R_y^{\mathcal{C}}$ et $R_z^{\mathcal{C}}$ de taille 4×4 .

Exemple. On note \mathcal{D} la configuration sur $\{1, 2, 7, 9\}$ en utilisant les relations d'ordre donnée précédemment. \mathcal{D} est égale à la configuration

$$\begin{array}{ll} 9 <_x 1 <_x 2 & \text{et} \quad 7 <_x 1 <_x 2 \\ 7 <_y 9 <_y 1 & \text{et} \quad 7 <_y 9 <_y 2 \\ 1 <_z 7 <_z 9 & \text{et} \quad 2 <_z 7 <_z 9 \end{array}$$

On obtient alors :

$$R_x^{\mathcal{D}} = \begin{pmatrix} 2 & -1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ -1 & -1 & 2 & 0 \\ -1 & -1 & 0 & 2 \end{pmatrix} \quad R_y^{\mathcal{D}} = \begin{pmatrix} 2 & 0 & 1 & 1 \\ 0 & 2 & 1 & 1 \\ -1 & -1 & 2 & -1 \\ -1 & -1 & 1 & 2 \end{pmatrix} \quad R_z^{\mathcal{D}} = \begin{pmatrix} 2 & 0 & -1 & -1 \\ 0 & 2 & -1 & -1 \\ 1 & 1 & 2 & -1 \\ 1 & 1 & 1 & 2 \end{pmatrix}$$

D'après la Proposition 3.3, une configuration est non-fixe si et seulement si il existe une extension linéaire de cette configuration qui est non-fixe. Ainsi nous allons calculer toutes les extensions linéaires jusqu'à ce qu'on en trouve une qui soit non-fixe.

Pour calculer les extensions linéaires d'une configuration \mathcal{C} sur \mathcal{E} , on remplace récursivement dans chacune des matrices $R_x^{\mathcal{C}}$, $R_y^{\mathcal{C}}$ et $R_z^{\mathcal{C}}$, chaque valeur égale à 0 par la valeur 1 puis par la valeur -1, à condition que ce remplacement respecte l'inégalité triangulaire. Plus précisément, pour tout couple $(i, j) \in \mathcal{E}^2$, si la valeur $R_x^{\mathcal{C}}(i, j)$ vaut zéro cela signifie que i et j ne sont pas comparable dans $<_x$ restreint à \mathcal{E} . Lorsque l'on souhaite remplacer la valeur de $R_x^{\mathcal{C}}(i, j)$ par la valeur 1, on souhaite rajouter dans $<_x$ restreint à \mathcal{E} la relation $i <_x j$. En rajoutant cette condition il faut veiller à ce que l'inégalité triangulaire ne soit pas violée, c'est-à-dire qu'il n'existe pas d'élément k de \mathcal{E} tel que $k <_x i$ et $j <_x k$. Avant de remplacer la valeur de $R_x^{\mathcal{C}}(i, j)$ par la valeur 1, il faut donc vérifier qu'il n'existe pas d'élément k de \mathcal{E} tel que $R_x^{\mathcal{C}}(k, i) = 1$ et $R_x^{\mathcal{C}}(k, j) = -1$.

Exemple. En remplaçant les 0 par des 1 ou des -1 en respectant l'inégalité triangulaire, on obtient par exemple les matrices suivantes :

$$\begin{pmatrix} 2 & -1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ -1 & -1 & 2 & -1 \\ -1 & -1 & 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 2 & 1 & 1 & 1 \\ -1 & 2 & 1 & 1 \\ -1 & -1 & 2 & -1 \\ -1 & -1 & 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 2 & -1 & -1 & -1 \\ 1 & 2 & -1 & -1 \\ 1 & 1 & 2 & -1 \\ 1 & 1 & 1 & 2 \end{pmatrix}$$

Ces matrices correspondent à l'extension linéaire de la configuration d'ordres \mathcal{D} sur $\{1, 2, 7, 9\}$ suivante :

$$\begin{array}{l} 7 <_x 9 <_x 1 <_x 2 \\ 7 <_y 9 <_y 2 <_y 1 \\ 1 <_z 2 <_z 7 <_z 9 \end{array}$$

Il ne nous reste plus qu'à voir comment déterminer si une extension linéaire est fixe ou non-fixe. Pour cela on se base sur le Théorème 3.11. Celui-ci nous indique qu'une extension linéaire sur \mathcal{E} est non-fixe si et seulement si il existe un ordre $<_i$ parmi les ordres $<_x$, $<_y$ et $<_z$ et un élément $e \in \mathcal{E}$ tels que e soit extrémal de $<_i$ restreint à \mathcal{E} et que les deux autres ordres restreint à $\mathcal{E} \setminus e$ forment une configuration non-fixe de deux ordres totaux sur $\mathcal{E} \setminus e$. Or d'après le Théorème 3.7, une configuration

de deux ordres totaux sur un ensemble \mathcal{E}' de taille trois est non-fixe si et seulement si il existe un élément de \mathcal{E}' qui n'est extrémal dans aucun des deux ordres. Cela revient à tester s'il existe un ordre $<_i$ parmi les ordres $<_x$, $<_y$ et $<_z$, un élément $e \in \mathcal{E}$ extrémal de $<_i$ restreint à \mathcal{E} et un élément a de $\mathcal{E} \setminus e$ tels que a ne soit extrémal dans aucun des deux autres ordres restreint à $\mathcal{E} \setminus e$.

Exemple. Dans cette extension linéaire, on a l'élément 7 qui est extrémal (minimal en l'occurrence) dans l'ordre $<_x$ restreint à $\{1, 2, 7, 9\}$. Or les restrictions de $<_y$ et $<_z$ à $\{1, 2, 9\} = \{1, 2, 7, 9\} \setminus 7$ sont égales à $9 <_y 2 <_y 1$ et $1 <_z 2 <_z 9$. L'élément 2 n'est extrémal dans aucune de ces deux restrictions, ainsi la configuration linéaire \mathcal{D} est non-fixe.

8.2 Caractérisation forte de classes

Dans la Section 6.2.2, nous avons vu que pour obtenir une discrimination forte d'une classe, nous commençons par trier les $(d+1)$ -uplets de points selon leur taux de discriminabilité, puis nous cherchons des paramètres pour discriminer cette classe et son complémentaire, chacun sous forme d'une "boule". Plus précisément, pour obtenir une discrimination forte d'une classe nous cherchons un sous-ensemble de $(d+1)$ -uplets de points \mathcal{D} (le plus petit possible pour plus de simplicité), un entier q strictement supérieur à la moitié de la taille de \mathcal{D} (le "rayon des deux boules" est alors égal à $\#\mathcal{D} - q$) et deux listes d'ensembles de signes T et \bar{T} ("centres des boules").

Or dans la Section 6.5, nous avons expliqué que nous calculons les paramètres d'une caractérisation forte du couple $(\mathcal{C}, \bar{\mathcal{C}})$ à partir d'une matrice binaire A . La liste d'ensembles de signes T est fixée avant de calculer cette matrice binaire. Chaque ligne de cette matrice correspond à un $(d+1)$ -uplet B de $\mathcal{B}(\mathcal{E})$ et chaque colonne correspond à un chirotope χ de \mathcal{M} . Dans cette matrice, le coefficient correspondant au $(d+1)$ -uplet B et au chirotope χ (noté $A_{\chi, B}$) est égal à 1 si et seulement si le chirotope χ serait mal classé par notre méthode en prenant $\mathcal{D} = \{B\}$.

La recherche des paramètres d'une caractérisation forte du couple $(\mathcal{C}, \bar{\mathcal{C}})$ revient à chercher sous-ensemble de lignes \mathcal{D} tel que la somme des coefficients de ces lignes soit strictement inférieure à $\#\mathcal{D}/2$ pour chaque colonne. Si on trouve un tel sous-ensemble de lignes, en notant max le maximum des sommes des colonnes, on prend alors $q = \#\mathcal{D} - max$. On rappelle que la liste \bar{T} s'obtient en prenant pour chaque $(d+1)$ -uplet B de \mathcal{D} le complémentaire de $T(B)$ (c'est-à-dire pour tout $B \in \mathcal{D}$, $\bar{T}(B) = \{-, 0, +\} \setminus T(B)$).

Pour faciliter la recherche des paramètres, on réduit le nombre de lignes de la matrice A en se restreignant aux $(d+1)$ -uplets dont le taux de discriminabilité est supérieur à une certaine valeur de seuil (généralement fixé à 0,7). En effet, les $(d+1)$ -uplets dont le taux de discriminabilité est faible correspondent dans la matrice à des lignes constituées de nombreuses valeurs égales à 1. De plus, on ordonne les lignes restantes de la matrice A selon le taux de discriminabilité des $(d+1)$ -uplets par ordre décroissant. On ordonne également des colonnes de la matrice selon le nombre de 1 que contiennent les colonnes par ordre décroissant.

Comme le maximum des sommes des colonnes max est strictement inférieur à $\#\mathcal{D}/2$, si $\#\mathcal{D} = 2k - 1$ la plus grande valeur que peut prendre max est $k - 1$ et si $\#\mathcal{D} = 2k$ la plus grande valeur que peut prendre max est également $k - 1$. On se contente alors de tester les ensembles de lignes de taille impaire. On cherche un ensemble \mathcal{D} le plus petit possible donc on va tester les ensembles par

taille croissante, ce qui revient à tester s'il existe un ensemble \mathcal{D} qui convienne de taille $2k - 1$ par valeur croissante de k .

Le cas $k = 1$ est très simple. En effet, il existe un sous-ensemble \mathcal{D} de taille 1 permettant d'obtenir une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$ si et seulement si il existe un $(d+1)$ -uplet B totalement discriminant entre \mathcal{C} et $\overline{\mathcal{C}}$. Or s'il existe un tel $(d+1)$ -uplet B , comme les lignes de A sont ordonnées par le taux de discriminabilité par ordre décroissant, cela signifie que la première ligne de A n'est composée que de la valeur 0. Dans ce cas, on prend alors $\mathcal{D} = \{B\}$ et $q = 1$.

Si aucun sous-ensemble de taille 1 ne convient, on procède de la façon suivante. D'après ce que nous avons vu, un sous-ensemble de lignes nous permet d'obtenir une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$ si et seulement si la somme des coefficients de ces lignes est inférieure ou égale à $k - 1$ pour chaque colonne. Ainsi pour qu'un sous-ensemble \mathcal{D} puisse convenir, il faut que pour la première colonne de A au moins $k - 1$ lignes de \mathcal{D} aient pour valeur 0. On teste alors chaque sous-ensemble de taille $2k - 1$ composé d'au moins $k - 1$ lignes dont la valeur est 0 pour la première colonne. Or nous avons ordonnées les lignes de A afin que les lignes ayant le plus de chance de nous servir pour constituer \mathcal{D} . Ainsi, on génère ces sous-ensembles en commençant par les sous-ensembles dont les lignes ont le plus petit indice.

Si l'on trouve un sous-ensemble \mathcal{D} nous permettant d'obtenir une caractérisation forte du couple $(\mathcal{C}, \overline{\mathcal{C}})$, la valeur de q est égale à $\#\mathcal{D} - \max$ où \max est le maximum des sommes des colonnes.

8.3 Calcul des symétries et asymétries

La plupart des animaux (les êtres humains y compris) présentent une symétrie bilatérale par rapport au plan médian sagittal. Il est donc important de savoir si des pathologies conservent cette symétrie ou au contraire crée des asymétries. Nous allons expliquer dans cette section le programme que nous avons mis en place permettant de calculer l'asymétrie d'une structure anatomique à partir du chirotope de l'ensemble de la structure.

Pour nos applications, on part d'une structure anatomique 3D qui possède un plan de symétrie (comme par exemple le cerveau). On suppose que pour chaque landmark p_i , le landmark symétrique de p_i par rapport au plan de symétrie appartient aussi à cet ensemble. On partitionne alors les landmarks en deux ensembles : la *demi-structure 1* formée des landmarks appartenant au plan de symétrie et des landmarks qui sont d'un côté du plan et la *demi-structure 2* qui est formée des landmarks appartenant au plan de symétrie et des landmarks qui sont de l'autre côté du plan. On dispose également du chirotope (noté χ) de l'ensemble des landmarks de la structure. Dans ce programme, on souhaite extraire de χ un chirotope χ_1 correspondant à la demi-structure 1. On souhaite également calculer, toujours à partir de χ , le chirotope χ_2 correspondant au symétrique de la demi-structure 2 afin de pouvoir le comparer ultérieurement à χ_1 .

Dans le cas général d'un ensemble de points en dimension d quelconque, cet ensemble de points est partitionné selon leur position par rapport à un hyperplan de \mathbb{R}^d .

L'idée du programme se base sur la propriété suivante : lorsque l'on symétrise un simplexe de \mathbb{R}^d (ensemble de $(d+1)$ -uplet de points) par rapport à un hyperplan de \mathbb{R}^d , alors on inverse l'orientation de ce simplexe. Ainsi si l'on dispose du chirotope d'un ensemble de points \mathcal{E} et qu'on sou-

haite obtenir le chirotope de l'ensemble de points \mathcal{E}' symétrique de \mathcal{E} par rapport à un hyperplan, il est plus rapide de déduire le chirotope de \mathcal{E}' en utilisant cette propriété plutôt que de calculer les coordonnées des points de \mathcal{E}' puis de calculer de façon classique le chirotope de \mathcal{E}' .

En notant m le nombre de points dans la demi-structure 1 (qui est également le nombre de points dans la demi-structure 2), chaque demi-structure est composée de $\binom{m}{d+1}$ $(d+1)$ -uplets de points. Les chirotopes χ_1 et χ_2 sont donc des vecteurs de taille $\binom{m}{d+1}$.

Pour chaque $(d+1)$ -uplet $B = (b_1, b_2, \dots, b_{d+1})$ de points de la demi-structure 1, on cherche dans χ le signe de B (ce qui se fait facilement grâce à la formule calculant l'indice d'un $(d+1)$ -uplet de la Section 7.2) et on recopie cette valeur dans χ_1 . On calcule ensuite le $(d+1)$ -uplet $B' = (b'_1, b'_2, \dots, b'_{d+1})$ formé par les points symétriques des points de B par rapport au plan. Or le signe de ce $(d+1)$ -uplet n'appartient pas forcément à χ . En effet, comme nous l'avons vu dans la définition du chirotope (voir Section 2.1.1), le chirotope est un vecteur contenant le signe des $(d+1)$ -uplets ordonnés (c'est-à-dire que le chirotope contient le signe du $(d+1)$ -uplet (u_1, \dots, u_{d+1}) si et seulement si pour tout $(i, j) \in \{1, \dots, d+1\}^2$ tels que $i < j$ on ait $u_i < u_j$). Par contre, nous avons vu dans la Section 2.1.1 que le chirotope est une application alternée, ce qui nous indique comment obtenir le signe de B' : en notant B'_{ord} le $(d+1)$ -uplet ordonné composé des mêmes points que B' , il faut calculer la permutation permettant de passer de B'_{ord} à B' et multiplier le signe de B'_{ord} par la signature de la permutation. On enregistre alors dans χ_2 l'opposé du signe de B' que nous venons de calculer puisque χ_2 est le chirotope correspondant au symétrique de la demi-structure 2.

Maintenant que nous avons vu comment programmer les principaux outils de notre méthode, nous allons tester notre méthode sur des jeux de données réels.



Partie V. Applications à la caractérisation de la morphologie 3D

Dans cette partie, nous présentons les résultats obtenus en appliquant notre méthode à des jeux de données anatomiques 3D. La première application (Chapitre 9) concerne l'anatomie comparée puisqu'il s'agit d'étudier la forme de crânes d'humains et de grands singes (bonobos et chimpanzés). Les deux autres applications se font sur des jeux de données cliniques. Il s'agit d'étudier respectivement les classes dentaires utilisées en orthodontie (Chapitre 10) et les déformations cranio-faciales présents chez des sujets atteints de craniosynostoses (Chapitre 11). Dans ce dernier chapitre nous étudions également la symétrie des crânes.

Méthodologie

Nous allons commencer par expliquer la méthodologie utilisée pour analyser des jeux de données composés des coordonnées de n landmarks en trois dimensions. Dans toutes les applications qui sont présentées dans cette partie, les landmarks sont en position générale. Ainsi tous les quadruplets de landmarks sont des bases. Notre méthode est constituée de trois étapes :

1) Calcul des chirotopes et distances entre les échantillons. Nous commençons par calculer pour chaque échantillon le chirotope associé à l'ensemble des n landmarks de cet échantillon, comme il est expliqué dans les Sections 2.1.1 et 7.1.1. En parallèle de ces calculs, nous déterminons pour chaque quadruplet s'il est fixe ou non en comparant chaque signe calculé avec le signe du premier chirotope.

Nous utilisons ensuite ces chirotopes pour calculer les distances entre chaque paire d'échantillons (Section 2.3). Ces distances nous servent à déterminer si un codage des échantillons à l'aide de chirotopes et l'utilisation de notre distance peut permettre de séparer ou de regrouper les échantillons en classes. Pour cela, nous commençons par présenter ces distances dans un tableau en utilisant un dégradé de couleur (vert pour la distance minimale entre deux échantillons et rouge pour la distance maximale). Ainsi nous pouvons dire si visuellement les classes de référence semblent homogènes ou hétérogènes et si elles semblent séparés. Toutes ces informations nous indiquent également en combien de sous-ensembles se partitionne l'ensemble des échantillons.

2) Classification automatique. Dans un deuxième temps, nous réalisons une classification automatique des échantillons en utilisant l'algorithme des k -means. Pour pouvoir utiliser cet algorithme, on doit donner comme paramètre le nombre de clusters à calculer. On utilise généralement comme paramètre le nombre de sous-ensembles d'échantillons que nous avons identifiés visuellement à l'aide du tableau des distances calculée précédemment. Comme nous l'avons expliqué dans la Section 5.1.2, les landmarks étant en position générale, nous représentons chaque chirotope par un point de $\{-1, 1\}^b$ où b est le nombre de quadruplets non-fixes. L'algorithme des k -means ne convergeant que vers un minimum local, nous le lançons 1 000 fois afin de conserver la meilleure partition possible (c'est-à-dire celle qui minimise le critère de convergence qui est la somme des distances au carré entre les points et le point moyen du cluster). Nous comparons alors les classes des experts avec les clusters composant la meilleure partition parmi les 1 000 calculées par l'algorithme. Nous calculons ensuite le *diamètre* de chaque cluster (c'est-à-dire la distance maximale entre deux chirotopes appartenant au cluster) et pour toutes paires de clusters nous calculons la *distance minimale* entre ces clusters (c'est-à-dire la distance maximale entre un chirotope appartenant au premier cluster et un chirotope appartenant au deuxième cluster). Ces calculs nous permettent de déterminer si les clusters fournis par l'algorithme des k -means sont homogènes et bien séparés.

diamètre
d'un
cluster

distance
minimale
entre deux
clusters

3) Caractérisation des classes. La dernière étape de notre analyse de jeux de données consiste à chercher un sous-ensemble de quadruplets permettant de caractériser des classes. Pour cela nous appliquons la méthode présentée dans le Chapitre 6. Nous obtenons alors pour chaque classe, un tri des quadruplets selon leur taux de discriminabilité. Nous cherchons ensuite parmi les quadruplets ayant le plus fort taux de discriminabilité, un sous-ensemble (le plus petit possible pour plus de simplicité) permettant d'obtenir une caractérisation forte ou à défaut une caractérisation faible (Section 6.2.2) de chaque classe \mathcal{C} de son complémentaire $\overline{\mathcal{C}}$. Dans une caractérisation forte nous cherchons des paramètres afin de définir cette classe et son complémentaire, chacun sous forme d'une "boule", alors que dans une caractérisation faible nous cherchons des paramètres afin de définir uniquement cette classe sous forme d'une "boule". Plus précisément, pour obtenir une discrimination forte d'une classe nous cherchons un sous-ensemble de $(d+1)$ -uplets de points \mathcal{D} (le plus petit possible pour plus de simplicité), un entier q strictement supérieur à la moitié de la taille de \mathcal{D} (le "rayon des deux boules" est alors égal à $\#\mathcal{D} - q$) et deux listes d'ensembles de signes T et \overline{T} ("centres des boules"). Dans le cas d'une discrimination faible d'une classe nous cherchons un sous-ensemble de $(d+1)$ -uplets de points \mathcal{D} (le plus petit possible pour plus de simplicité), un entier l strictement inférieur à la moitié de la taille de \mathcal{D} (le "rayon de la boule") et une liste d'ensembles de signes T ("centre de la boule").

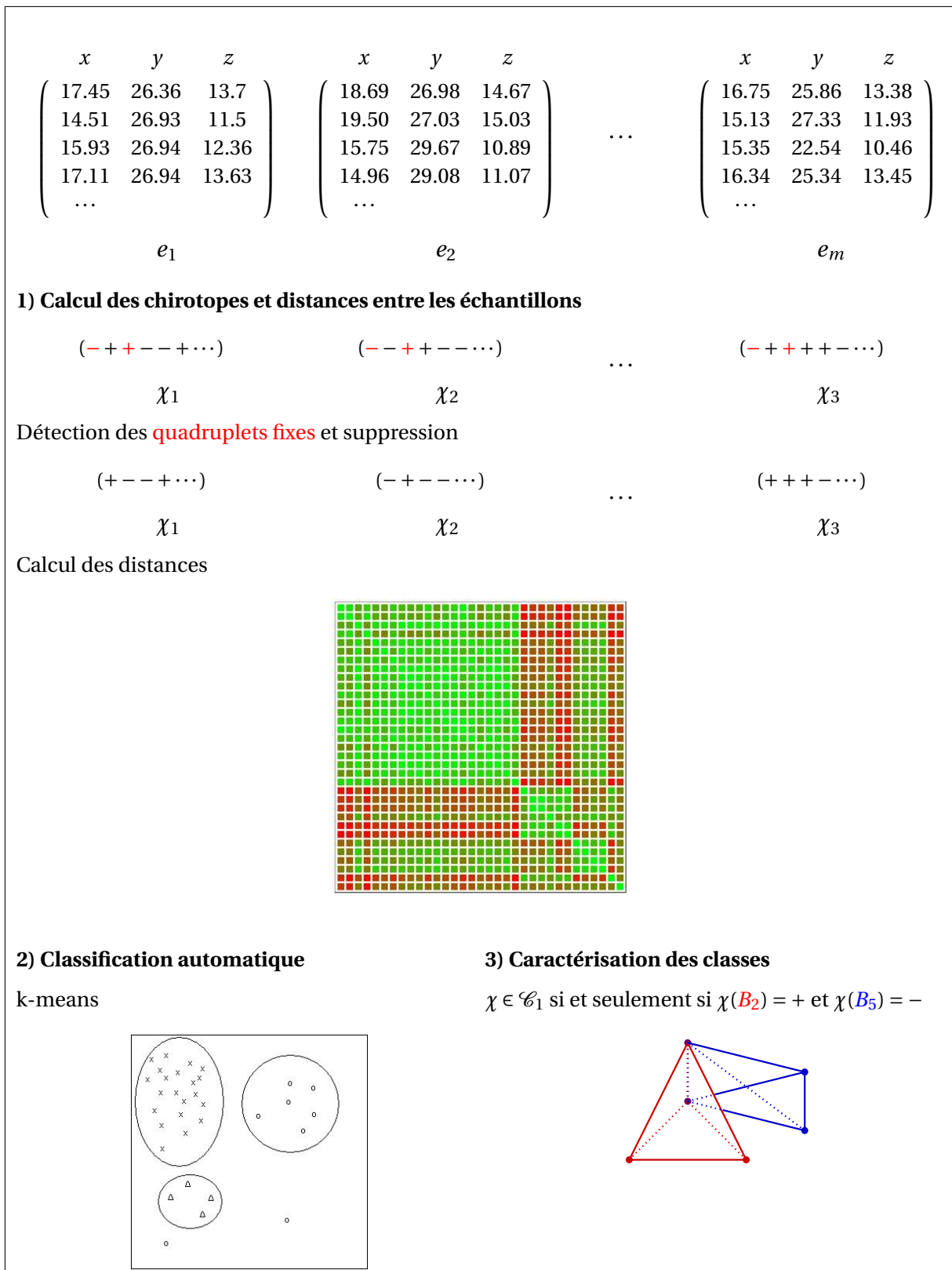


FIGURE 8.4 – Schéma de la méthodologie

Anatomie comparée : analyse des différences humains / grands singes

Dans ce chapitre, nous allons étudier les résultats que nous avons obtenus sur une base de données de crânes d'humains et de deux espèces de singes : les chimpanzés communs et les bonobos (appelés aussi chimpanzés nains). Pour plus de facilité on utilisera le terme de chimpanzé pour parler de chimpanzé commun. Nous avons travaillé en collaboration avec José Braga, professeur d'anthropologie à l'Université Toulouse 3, sur cette base de donnée composée des coordonnées en trois dimensions de 16 landmarks relevés sur 182 crânes humains, 58 crânes de bonobos et 50 crânes de chimpanzés. Ces 16 landmarks ont été pris sur la face et la base du crâne comme l'indique la Figure 9.1.

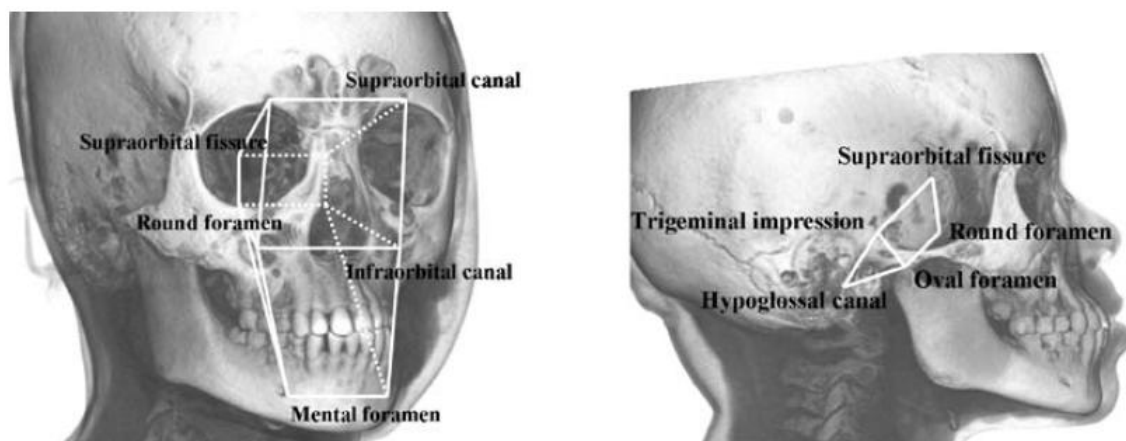


FIGURE 9.1 – Position des 16 landmarks (image provenant de [8])

Pour les humains nous disposons de 68 mesures prises sur des enfants dont les âges vont de quelques jours à 12 ans, 59 mesures prises sur des adolescents qui ont entre 12 ans et 18 ans et 55 mesures prises sur des adultes. Ces données provenant d'humains ont déjà été étudiés pour estimer l'âge du squelette (voir l'article [8]).

Pour ce qui est des singes, nous avons 58 mesures prises sur des bonobos et 50 sur des chimpanzés communs. L'âge de chaque singe a été approximé à partir de l'état dentaire et séparé en différentes classes : les nouveaux nés (notés NJ), les jeunes (notés J) et les adultes (notés A).

9.1 Calcul des chirotopes et distances entre les échantillons

Afin d'analyser ces données, nous avons calculé le chirotipe associé à chaque échantillon. Chaque chirotipe est composé de $\binom{16}{4} = 1820$ signes. Les landmarks étant en position générale, l'ensemble des landmarks de chaque échantillon forme 1 820 bases. En comparant ces chirotipes nous avons trouvé que 1 241 bases sont fixes, soit environ 68% du nombre total de bases. Ainsi les comparaisons entre les chirotipes ne se feront que sur 579 bases.

Distances moyennes entre les espèces. Nous avons calculé les distances entre tous les chirotipes. La Figure 9.2 représente la moyenne des distances entre les chirotipes en fonction des espèces. Ces moyennes sont comptabilisées en pourcentage du nombre de bases fixes. L'échelle de couleurs est calculée sur l'ensemble des chirotipes, ce qui indique que la plus grande distance entre deux chirotipes est d'environ 56% des bases non-fixes et que la distance minimale est d'environ 2% des bases non-fixes).

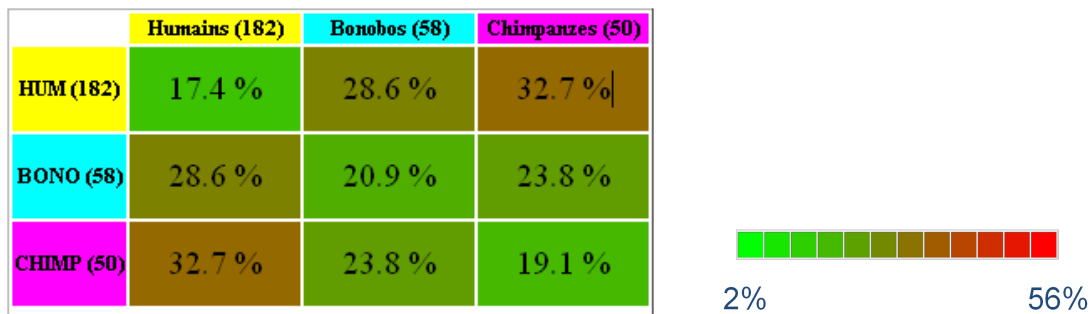


FIGURE 9.2 – Moyenne des distances entre les chirotipes en fonction des classes définies par les experts (en pourcentage du nombre de bases non-fixes). Les valeurs entre parenthèse dans le tableau indiquent l'effectif des classes. L'échelle de couleurs est calculée sur l'ensemble des chirotipes.

Dans ce tableau on remarque que les moyennes les plus faibles correspondent aux moyennes des distances entre deux humains et aux moyennes des distances entre deux singes. Cela signifie qu'en moyenne, on arrive à distinguer les chirotipes correspondant à des crânes d'humains des chirotipes correspondant à des crânes de singes. On remarque également que la moyenne des distances entre les chirotipes de bonobos et les chirotipes de chimpanzés est légèrement plus grande que la moyenne des distances entre les chirotipes d'une même espèce. Il semblerait donc possible de distinguer les bonobos des chimpanzés à l'aide de chirotipes définis sur ces ensembles de landmarks.

Nous avons également étudié ces données en fonction de l'âge des sujets. Le tableau de la Figure 9.3 représente la moyenne des distances entre les chirotipes en fonction des espèces et de

		Humains			Bonobos			Chimpanzes		
		Enfants (68)	Adolescents (59)	Adultes (55)	NJ (3)	Jeunes (29)	Adultes (26)	NJ (2)	Jeunes (25)	Adultes (23)
HUM	ENF (68)	5.65 %	5.65 %	5.62 %	9.96 %	8.6 %	8.67 %	7.8 %	9.28 %	11 %
	ADO (59)	5.65 %	5.28 %	5.35 %	11 %	9.55 %	9.25 %	8.5 %	10.1 %	11.7 %
	ADU (55)	5.62 %	5.35 %	5.4 %	10.6 %	9.23 %	9.03 %	8.27 %	9.75 %	11.4 %
BONO	NJ (3)	9.96 %	11 %	10.6 %	6.48 %	5.95 %	8.23 %	7.96 %	7.65 %	8.85 %
	J (29)	8.6 %	9.55 %	9.23 %	5.95 %	5.18 %	7.18 %	7.38 %	6.88 %	8.55 %
	A (26)	8.67 %	9.25 %	9.03 %	8.23 %	7.18 %	7 %	7.22 %	7.05 %	7.74 %
CHIMP	NJ (2)	7.8 %	8.5 %	8.27 %	7.96 %	7.38 %	7.22 %	6.43 %	6.42 %	7.43 %
	J (25)	9.28 %	10.1 %	9.75 %	7.65 %	6.88 %	7.05 %	6.42 %	5.94 %	6.27 %
	A (23)	11 %	11.7 %	11.4 %	8.85 %	8.55 %	7.74 %	7.43 %	6.27 %	5.54 %

FIGURE 9.3 – Moyenne des différences entre les espèces en fonction de l'âge. Les valeurs entre parenthèse dans le tableau indiquent l'effectif. L'échelle de couleurs est la même que celle de la Figure 9.2

l'âge. Ce tableau nous indique que l'écart entre les crânes humains reste faible quel que soit le stade de la croissance puisque la moyenne des distances entre deux crânes humains est d'environ 5,5% et ce quel que soit l'âge. De même, d'après ce tableau, les crânes de chimpanzés les plus proches des crânes humains sont ceux des nouveaux nés (NJ). Ces deux résultats nous ont été confirmés par José Braga. En effet, le crâne de l'Homme conserve sa forme juvénile au stade adulte, contrairement aux crânes des grands singes (chimpanzés, gorilles) qui se modifient de manière beaucoup plus importants. Par contre, curieusement, on ne retrouve pas ce résultat entre les humains et les bonobos.

9.2 Classification automatique

Classification automatique en deux clusters. Nous avons cherché à obtenir deux clusters par l'algorithme des k -means pour tenter de séparer les Humains des Singes comme nous l'avons remarqué visuellement dans la matrice des distances. Les clusters fournis par l'algorithme des k -means sont présentés par la matrice de confusion de la Table 9.1. Seuls quatre chirotopes sont mal-classés par l'algorithme des k -means, deux sont des chirotopes d'humains et deux sont des chirotopes de

singes. Ainsi, près de 99% des chirotopes sont bien classés par cet algorithme simple. On arrive donc à distinguer presque parfaitement les crânes d'humains des crânes de singes.

	Cluster 1	Cluster 2
Humains	180	2
Singes	2	106

TABLE 9.1 – Matrice de confusion des deux clusters fournis par l'algorithme des k -means.

Classification automatique en trois clusters. Nous avons utilisé la même méthode afin d'obtenir trois clusters pour voir si on pouvait retrouver les trois espèces (Humains / Bonobos / Chimpanzés). Les clusters fournis par l'algorithme des k -means sont présentés par la matrice de confusion de la Table 9.2. Le cluster 1 est constitués de 178 chirotopes dont 176 sont des chirotopes d'humains, un chirotope est un chirotope de bonobo et un chirotope est un chirotope de chimpanzé.

Cette matrice nous indique qu'au total, 29 chirotopes sont mal-classés par l'algorithme des k -means, soit 10% des chirotopes. A noter que les chirotopes de bonobo et de chimpanzé qui sont classés dans le cluster 1 sont les deux chirotopes de singes qui étaient mal-classés dans la Table 9.1.

	Cluster 1	Cluster 2	Cluster 3
Humains	176	6	0
Bonobos	1	44	13
Chimpanzés	1	8	41

TABLE 9.2 – Matrice de confusion entre les 3 espèces et les 3 clusters fournis par l'algorithme des k -means.

Diamètres et distances minimales. Nous avons étudié plus en détail les chirotopes afin de déterminer si la partition des chirotopes selon les espèces est satisfaisante ou non, c'est-à-dire si les ensembles de chirotopes correspondant aux espèces sont homogènes et s'ils sont bien différenciés. Pour savoir si ces ensembles sont homogènes, nous avons calculé le diamètre de chaque espèce, c'est-à-dire la distance maximale entre deux chirotope d'une même espèce. Les résultats sont présentés dans la Table 9.3. Nous avons également calculé les distances minimales entre les différentes espèces (voir Table 9.4). Il apparaît alors que les distances minimales entre les espèces sont nettement inférieures aux diamètres des espèces. Ainsi, que ce soit en séparant les chirotopes en deux ensembles "Humains / Singes" ou en trois ensembles "Humains / Bonobos / Chimpanzés", les ensembles de chirotopes correspondant aux espèces ne fournissent pas une partition satisfaisante des chirotopes.

9.3 Caractérisation des différentes espèces

Nous avons appliqué la méthode présentée dans le Chapitre 6 afin d'obtenir pour chaque espèce un classement des bases selon leur taux de discriminabilité. Les plus grands taux de discriminabi-

Espèce	Diamètre
Humains	38,7%
Singes	42,0%
Bonobos	41,5%
Chimpanzés	36,1%

TABLE 9.3 – Diamètres (i.e. distances maximales entre deux chirotopes d’une même espèce) des différentes espèces. Ce diamètre est donné en pourcentage du nombre de bases non-fixes.

	Humains	Bonobos
Bonobos	12,3%	
Chimpanzés	13,6%	9,3%

TABLE 9.4 – Distances minimales entre les chirotopes des espèces (en pourcentage du nombre de bases non-fixes).

lité entre les humains et les non-humains (donc entre les humains et les singes) sont de 0,97. La Figure 9.4 représente la base ayant le plus fort taux de discriminabilité entre les humains et les singes. Les quatre landmarks composant cette base sont en rouge sur la figure. Il s’agit du *left supraorbital canal*, du *right trigeminal impression*, du *left round foramen* et du *right hypoglossal canal*. Notons B_1 cette base. Tous les humains sauf un possèdent pour B_1 la même orientation que la figure de gauche et tous les singes sauf trois ont pour B_1 la même orientation que la figure de droite. Ainsi, on remarque sur cette figure que la majorité des humains ont le landmark du canal supra-orbitaire gauche d’un côté du plan formé par les trois autres landmarks de cette base (plan qui est représenté en vert sur la Figure 9.4) alors que ce landmark est de l’autre côté du plan pour la plupart des singes.

Par contre, bien que les plus grands taux de discriminabilité entre les humains et les singes soient de 0,97, il n’existe ni caractérisation forte ni caractérisation faible (voir Section 6.2.2) de l’ensemble des chirotopes correspondant aux humains.

Le seul chirotope d’humain n’ayant pas pour B_1 la même orientation que la partie gauche de la Figure 9.4 faisait partie des chirotopes mal classés par les k -means lorsque nous cherchions à obtenir deux clusters. Nous avons représenté dans la Figure 9.6 une vue de profil de l’ensemble des landmarks de cet humain. La Figure 9.5 représente les landmarks du chirotope le plus proche du point moyen de l’ensemble des chirotopes d’humains et la Figure 9.7 représente les landmarks du chirotope le plus proche du point moyen de l’ensemble des chirotopes de singes. On observe bien que la position des landmarks pour l’humain mal classé (Figure 9.6) est plus proche de la position du “singe moyen” (Figure 9.7) que de celle de “l’humain moyen” (Figure 9.5).

Que l’on considère soit les chimpanzés soit les bonobos, les plus grands taux de discriminabilité entre une de ces espèces et les autres sujets sont de 0,76. Encore une fois il n’existe ni caractérisation forte (voir Section ??), ni de caractérisation faible (Section ??) de l’ensemble des chirotopes correspondant aux chimpanzés ou au bonobos.

Ces résultats nous indiquent qu’en appliquant notre méthode à ces données, on parvient à séparer les crânes d’humains des crânes de singes (99% de réussite). Notre méthode nous permet

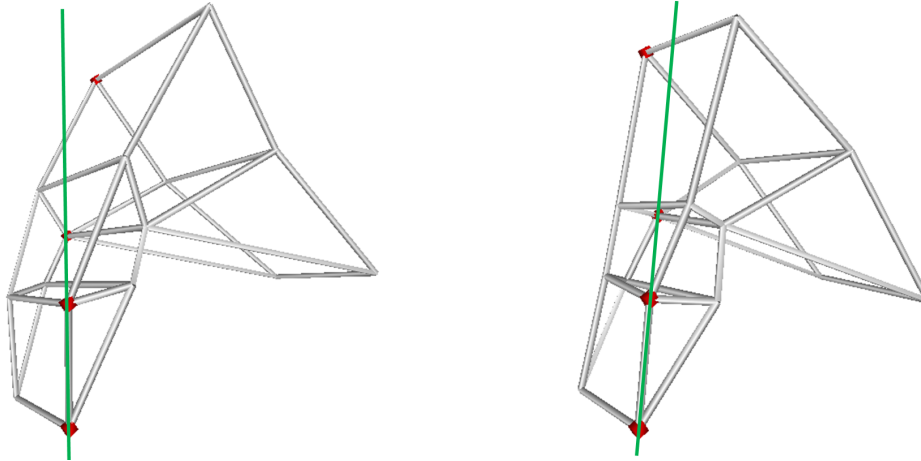


FIGURE 9.4 – Représentation de la base ayant le plus fort taux de discriminabilité entre les humains et les singes. Tous les humains sauf un possèdent pour cette base la même orientation que la figure de gauche et tous les singes sauf trois ont pour cette base la même orientation que la figure de droite.

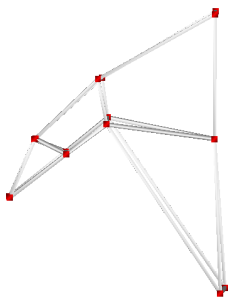


FIGURE 9.5 – Vue de profil de “l’humain moyen”, c’est-à-dire de l’humain dont le chirotope est le plus proche du point moyen de l’ensemble des chirotopes d’humains.

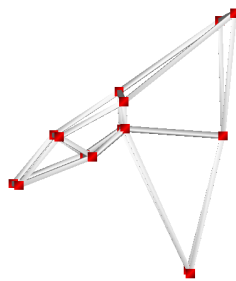


FIGURE 9.6 – Vue de profil de l’humain n’ayant pas pour B_1 la même orientation que la partie gauche de la Figure 9.4. Cet humain était également mal-classé par l’algorithme des k -means lorsque nous cherchions à obtenir deux clusters.

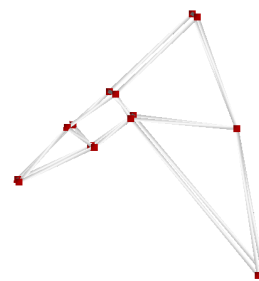


FIGURE 9.7 – Vue de profil du “singe moyen”, c’est-à-dire du singe dont le chirotope est le plus proche du point moyen de l’ensemble des chirotopes de singes.

également de séparer ces données selon les espèces (humains, bonobos, chimpanzés) avec un taux de réussite de 90%. Par contre un petit nombre de crânes d'humains et de singes nous a empêché de trouver une caractérisation des espèces. Il serait intéressant d'étudier ces crânes pour essayer d'expliquer ce phénomène.

Orthodontie : analyse de classes dentaires

Présentation du problème clinique. L'orthodontie est une discipline de la médecine-dentaire, qui se consacre au diagnostic et à la correction des mauvaises positions des dents et des mâchoires. Le but de l'orthodontie est de corriger l'occlusion dentaire qui est la façon dont les dents de la mandibule (qui est mobile) rentrent en contact avec les dents du maxillaire (fixées au crâne). A la fin du dix-neuvième siècle, E.H. Angle (voir [55]) a réparti les variations d'occlusion dans le sens antéro-postérieur en trois classes (classe I, classe II et classe III).

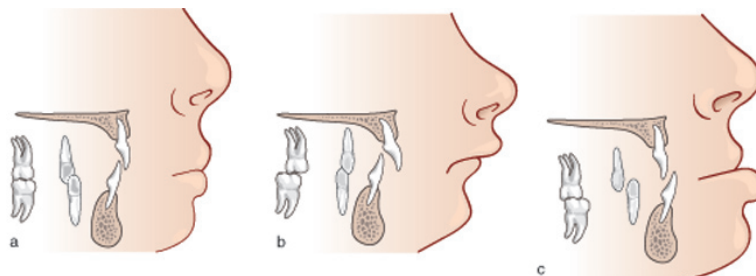


FIGURE 10.1 – a. Classe I (occlusion normale). b. Classe II. c. Classe III. (image provenant de l'Université Médicale Virtuelle Francophone)

Bien que d'autres moyens de diagnostic ont été définis, ces classes sont toujours utilisées par les orthodontistes ([26, 63, 79]). Pour savoir à quelle classe appartient un individu, on étudie la position de la première molaire inférieure par rapport à la première molaire supérieure (voir Figure 10.1). La classe I contient les individus dont la position de ces dents est normale, la classe II contient les individus dont la molaire inférieure est en arrière de la position normale et la classe III contient les individus dont la molaire inférieure est en avant de la position normale.

Dans [79], Treil et al. ont proposé une nouvelle approche de traitement orthodontique fondée sur l'utilisation de dix landmarks anatomiques. Pour cela, ils ont utilisé une base de données composée d'images scanner en trois dimensions (CT-Scan) de crânes de 305 individus répartis dans les trois classes d'Angle (125 dans la classe I, 93 dans la classe II et 87 dans la classe III). Les experts

ont pointé sur ces images les coordonnées 3D de dix landmarks anatomiques (voir Figure 10.2). Six landmarks ont été pris sur la face. Il s'agit des foramen supra-orbitaires (RSO et LSO), foramen infra-orbitaire (RIO et LIO) et foramen mentonniers (RM et LM). Les quatre autres landmarks correspondent aux foramen mandibulaires (RMd et LMd) et à la tête des deux marteaux situés dans l'oreille moyenne (RHM et LHM). Chacune des classes d'Angle a été divisée en trois sous-classes (normales, hypodivergentes et hyperdivergentes) qui traduisent l'écartement des points de la face sur l'axe vertical.

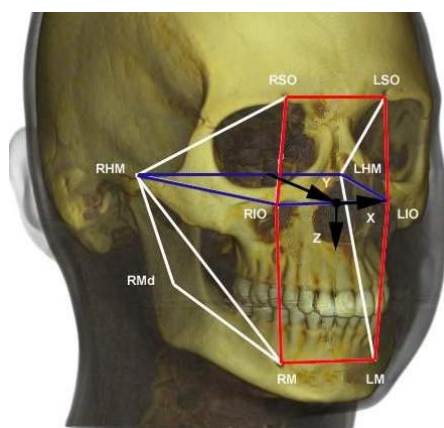


FIGURE 10.2 – Position des 10 landmarks (image fournie par Treil)

Afin d'analyser ces données, nous avons calculé le chirotope associé à chaque échantillon. Chaque chirotope est composé de $\binom{10}{4} = 210$ signes correspondant à 210 bases (puisque les landmarks sont en position générale). En comparant ces chirotopes nous avons trouvé que 162 bases sont fixes, soit environ 77% du nombre total de bases. Ainsi les comparaisons entre les chirotopes se feront sur les 48 bases non-fixes.

Nous avons déjà utilisé ces données pour illustrer les bases fixes issues d'ordres sur les coordonnées des points (Section 3.1.2 et 3.3.3). Après avoir défini des ordres sur les coordonnées des landmarks, nous avons déterminé que 20 bases parmi les 210 étaient fixes en utilisant uniquement ces ordres.

Distances moyennes entre les classes. Nous avons calculé les distances entre tous les chirotopes en pourcentage du nombre de bases non-fixes. La distance minimale entre deux chirotopes est égale à 0 (c'est-à-dire que deux chirotopes sont identiques) et la distance maximale est égale à 78% du nombre total de bases non-fixes. Ces valeurs nous ont permis de créer une échelle de couleur allant du vert (peu de différences) au rouge (beaucoup de différences). La Figure 10.3 représente la moyenne des distances entre les chirotopes en fonction des classes définies par les experts.

Dans ce tableau on remarque que les trois zones correspondant aux trois moyennes les plus petites se distinguent clairement. De la plus petite moyenne à la plus grande moyenne, ces zones correspondent à la moyenne entre deux chirotopes de la classe III, la moyenne entre un chirotope de la classe I et un chirotope de la classe III et la moyenne entre deux chirotopes de la classe I.

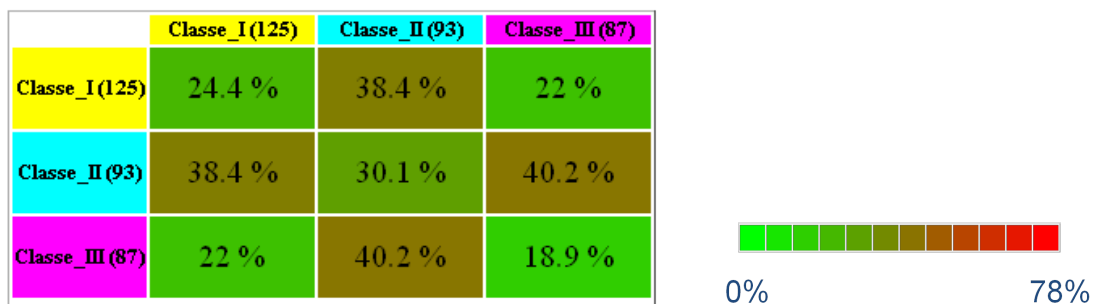


FIGURE 10.3 – Moyenne des distances entre les chirotopes en fonction des classes définies par les experts (en pourcentage du nombre de bases non-fixes). Les valeurs entre parenthèse dans le tableau indiquent l'effectif des classes.

On remarque également qu'en moyenne, la distance entre deux chirotopes de la classe II est bien inférieure à la distance entre un chirotope de la classe II et un chirotope d'une autre classe. Ainsi la classe II semble bien différenciée des autres classes par notre méthode.

Répartition des chirotopes en deux parties. Les résultats donnés par la Figure 10.3 semble nous indiquer que l'ensemble des chirotopes se partitionne naturellement en deux parties, la première contenant les chirotopes des classes I et III et la deuxième contenant les chirotopes de la classe II. Afin de déterminer si ces deux parties sont homogènes et bien séparées, nous avons alors calculé leurs diamètres (c'est-à-dire pour chaque partie, la distance maximale entre deux chirotopes de cette partie) et la distance minimale entre ces deux parties. Il apparait alors que les diamètres de ces classes sont égaux à 64.8% pour la partie constituée des chirotopes des classes I et III et à 66.7% pour la partie constituée des chirotopes de la classe II. Ces valeurs ne sont pas très éloignées de la distance maximale entre deux chirotopes qui est de 78%, ce qui indique que ces parties ne semblent pas homogènes. Ces deux parties ne sont pas non plus éloignées. En effet, nous avons vu que deux chirotopes sont identiques et ces deux chirotopes correspondent respectivement à un crâne d'un individu de la classe I et à celui d'un individu de la classe II.

Classification automatique en deux clusters. Nous voulions déterminer si les deux parties précédentes ont des grands diamètres et si la distance minimale entre ces deux parties est égale à zéro, à cause de données aberrantes. Nous avons alors utilisé l'algorithme des k -means afin d'obtenir une partition des chirotopes en deux clusters. Les clusters fournis par l'algorithme des k -means sont présentés par la matrice de confusion de la Table 10.1. Ces deux clusters correspondent aux deux parties sauf pour 27 chirotopes (soit moins de 9% des données). Parmi ces 27 chirotopes mal classés par l'algorithme des k -means, 20 chirotopes correspondent à des crânes de la classe II qui sont donc classés par l'algorithme avec les chirotopes des classes I et III. Les 7 autres chirotopes mal classés par l'algorithme appartiennent à la classe I et sont classés par l'algorithme avec les chirotopes des classes II. Ainsi aucun chirotope de la classe II n'est classé par l'algorithme avec les chirotopes des classes II.

	Cluster 1	Cluster 2
Classes I et III	205	7
Classe II	20	73

TABLE 10.1 – Matrice de confusion entre les classes d'Angle et les deux clusters retournés par l'algorithme des k -means.

Bases discriminantes entre ces deux parties. Comme il existe deux chirotopes identiques dont l'un correspondant à un crâne d'un individu de la classe I et l'autre à celui d'un individu de la classe II, aucune base n'est totalement discriminante et il n'est pas possible de trouver une caractérisation de ces parties. Il apparaît que les quatre bases présentant le plus haut taux de discriminabilité (entre 0,85 et 0,89) sont formées de la même façon. Ces quatre bases sont composés du foramen infra-orbitaire droit (RIO dans la Figure 10.2), du foramen infra-orbitaire gauche (LIO), de l'un des deux foramen supra-orbitaires (RSO ou LSO) et de l'un des deux foramen mandibulaires (RM ou LM). La Figure 10.4 présente l'une de ces bases.

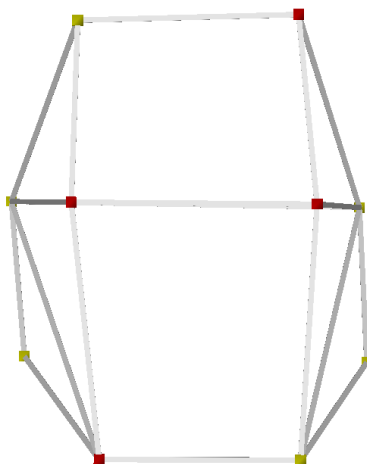


FIGURE 10.4 – Une des quatre bases présentant le plus haut taux de discriminabilité (en rouge).

Discussion. En codant les ensembles de landmarks par des chirotopes, on parvient à séparer les crânes appartenant aux classes I et III des crânes appartenant à la classe II (plus de 91% de réussite). En étudiant les bases discriminantes entre les chirotopes de la classe II et les chirotopes des autres classes, on observe un changement de convexité de la face. Ainsi on en déduit que dans la majorité des cas, les crânes de la classe II ont une convexité faciale tandis que les crânes des autres classes présentent une concavité faciale.

Jacques Treil nous a confirmé qu'en règle générale, les individus appartenant à la classe I ont une légère concavité faciale, que les individus de la classe II ont une convexité faciale le plus souvent par rétrognathie (c'est-à-dire un recul du maxillaire inférieur) tandis que les individus de la classe III ont une concavité faciale le plus souvent par prognathie (avancée du maxillaire inférieur).

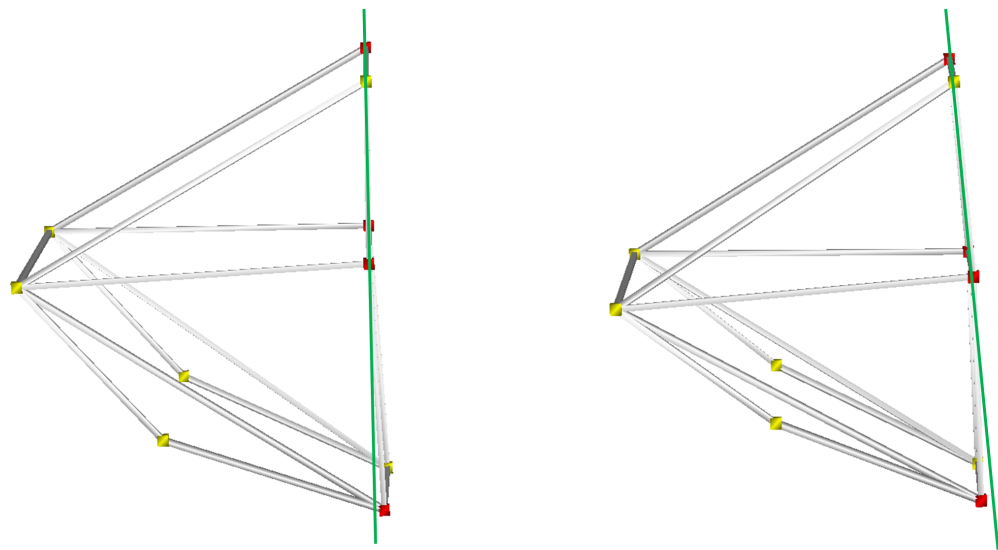


FIGURE 10.5 – Une des quatre bases présentant le plus haut taux de discriminabilité (en rouge).

Par contre, comme aucune base n'est totalement discriminante, on peut en conclure qu'une occlusion normale (classe I) ne se traduit pas forcément par une concavité faciale et que le fait que les dents du bas plus décalées en arrière (classe II) ne se traduit pas forcément par un recul du maxillaire inférieur.

De plus, en étudiant les modèles numériques tridimensionnels des différents crânes, on remarque que de nombreux individus souffrent de lourdes déformations crâniennes (voir Figure 10.6). Pour certains individus, ces déformations ont pour conséquence que les deux foramen mandibulaires (RM et LM) ne sont pas du même côté d'un plan formé par trois des quatre foramen infra-orbitaires (RIO et LIO) et supra-orbitaires (RSO et LSO). Ces déformations peuvent également expliquer la présence de chirotopes mal-classés et donc l'impossibilité de discriminer la classe II des deux autres classes.



FIGURE 10.6 – Exemple d'un modèle numérique tridimensionnel d'un crâne présentant de lourdes déformations crâniennes (images fournies par Jacques Treil).

Déformation cranio-faciale : analyse de craniosynostoses

Présentation du problème clinique

La craniosynostose (ou craniosténoses) est une pathologie rare (environ 1 naissance sur 2 500) touchant les enfants dans laquelle une ou plusieurs sutures du crâne se soudent prématurément. Les enfants présentant une craniosynostose souffrent d'une déformation du crâne (voir Figure 11.1). Pour corriger cela, ils doivent subir des interventions chirurgicales quelquefois très lourdes.



FIGURE 11.1 – Photographies d'un enfant présentant une craniosynostose coronale avant (à gauche) et après (à droite) intervention chirurgicale (photographies provenant du site internet du *International Craniofacial Institute*).

On s'intéressera ici à la craniosynostose coronale dans laquelle uniquement la suture coronale (également appelé suture fronto-pariétale) est fusionnée. La Figure 11.2 représente l'emplacement de cette suture. On peut distinguer trois types de craniosynostose coronale : le type "Left Uni-coronal Craniosynostosis" (LUCS) et le type "Right Uni-coronal Craniosynostosis" (RUCS) pour lesquels un seul côté est affecté et le type "Bilateral Coronal Craniosynostosis" (BCS) pour lequel les deux côtés sont fusionnés.

Comme il est rappelé dans [31], les experts pensent que la déformation du crâne serait due à une compensation de la croissance du cerveau qui est contrainte par la fusion de la suture. Récemment, certains modèles de souris ont démontré que la déformation du crâne peut se produire avant la fermeture de la suture, ce qui suggère que les deux phénomènes pourraient résulter d'une mutation génétique commune. Pour aller plus loin, il est alors indispensable d'utiliser des outils morphométriques tridimensionnels afin de quantifier et d'analyser précisément la déformation de la forme du crâne dans cette pathologie.

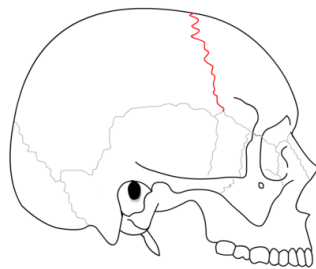


FIGURE 11.2 – Vue latérale du crâne dans laquelle la suture coronale est représentée en rouge (Rosario van Tulpe, via Wikipedia).

Dans [31], les auteurs ont utilisé des méthodes classiques (Procuste et Analyse en Composantes Principales) basées sur des landmarks pour étudier une base de données du Consortium international Craniosynostosis et évaluer l'asymétrie de la pathologie.

Cette base de données cliniques est composée d'images scanner en trois dimensions (CT-Scan) de crânes de 40 enfants présentant une craniosynostose coronale mais ne présentant pas d'autres signes ou symptômes. Parmi ces 40 enfants, 8 présentent uniquement une fusion sur la partie gauche du crâne (LUCS=8), 17 présentent uniquement une fusion sur la partie droite du crâne (LUCS=17) et 15 ont une suture bilatérale (BCS=15). Cette base de données contient également les images scanner en trois dimensions de crânes de 20 individus non affectés (Unaffected = 20) du même âge. Des experts en anatomie ont récupéré sur ces images les coordonnées 3D de 41 landmarks et ont ajouté 92 semi-landmarks le long de courbes prédéfinies, comme illustré sur la Figure 11.3. Aucune mise à l'échelle, rotation ou translation n'a été effectuée.

Dans le cadre d'une collaboration avec The Pennsylvania State University et The Richtsmeier lab, nous avons pu tester notre nouvelle méthode morphométrique sur les mêmes données et nous proposons dans cette section quelques résultats. Nous commencerons par présenter nos résultats portant sur une étude de l'ensemble des landmarks (landmarks anatomiques et semi-landmarks), puis nous présenterons les résultats que nous avons obtenus en analysant uniquement les landmarks anatomiques. Nous présenterons enfin les résultats de notre étude de l'asymétrie du crâne et nous comparerons ces résultats à ceux obtenus par Heuzé et al. dans [31].

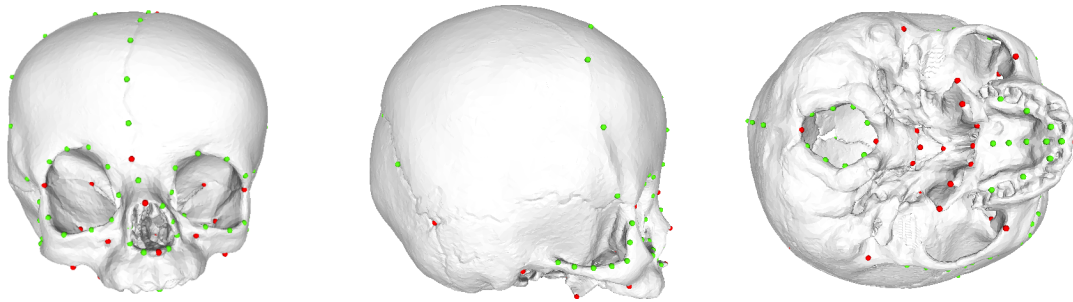


FIGURE 11.3 – Illustration des 133 landmarks tridimensionnels (le crâne présenté ici est un crâne RUCS). Les 41 landmarks anatomiques sont en rouge et les semi-landmarks curvilignes sont affichés en vert.

11.1 Etude sur l'ensemble des landmarks

Afin d'analyser ces données, nous avons calculé pour chaque échantillon le chirotope du matroïde orienté associé à cet échantillon. Chaque chirotope est composé de $\binom{133}{4} = 12\,457\,445$ signes. Les points étant en position générale, chaque échantillon est donc composé de 12 457 445 bases. Le calcul pour les 60 individus a pris environ six heures sur un ordinateur standard mais n'est effectué qu'une fois. En comparant ces chirotopes nous avons trouvé que 9 781 446 bases sont fixes, soit environ 79% du nombre total de bases. Ainsi la comparaison entre les chirotopes ne se fera que sur 2 675 999 bases.

Les résultats que nous allons aborder ici ont été présentés en partie dans [23] et [24].

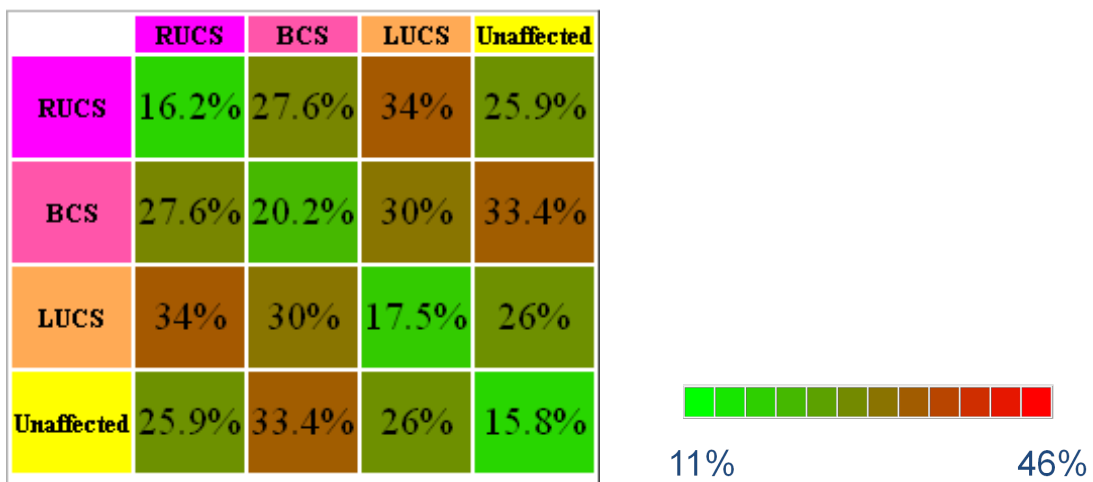


FIGURE 11.4 – Moyenne des distances entre les chirotopes en fonction des classes définies par les experts (en pourcentage du nombre de bases non-fixes).

Distance entre les chirotopes. Nous avons alors calculé les distances pour chaque paire d'échantillons. La Figure 11.4 représente la moyenne des distances entre les chirotopes en fonction des classes définies par les experts. Ces moyennes sont comptabilisées en pourcentage du nombre de bases fixes. L'échelle de couleur est faite sur les distances entre toutes les paires de chirotopes. Ainsi la distance la plus faible entre deux chirotopes est d'environ 11% des bases non-fixes et la distance la plus grande est d'environ 46%. Dans la Figure 11.4 on remarque que les moyennes les plus faibles sont entre les chirotopes associés à des crânes d'une même classe. De plus, la moyenne la plus élevée est entre les chirotopes des RUCS et les chirotopes des LUCS, suivie de près par la moyenne entre les chirotopes des BCS et les chirotopes des Unaffected.

Pour continuer l'analyse des distances entre les chirotopes, nous avons représenté la matrice des distances sous forme d'un tableau de couleurs (Figure 11.4). Ce tableau utilise le même dégradé de couleurs que celui de la Figure 11.4. Ainsi, plus une case est verte claire et plus la distance entre les deux chirotopes est faible alors que plus une case est rouge et plus la distance entre les chirotopes est importante. Lorsque l'on étudie la matrice des distances entre chaque paire de chirotopes, on observe que des carrés verts se distinguent dans ce tableau. Ces carrés sont situés sur la diagonale du tableau et correspondent à des distances entre deux chirotopes d'une même classe. Comme ce sont les seules zones contenant des cases vertes claires, ce premier résultat semble indiquer que la fusion des sutures est bien corrélée à des changements au niveau de la position relative des landmarks.

Classification automatique. Pour confirmer l'impression visuelle donnée par la Figure 11.5 nous avons cherché à obtenir quatre clusters par l'algorithme des k -means. Les clusters fournis par l'algorithme des k -means correspondent exactement aux classes des experts. Nous avons également utilisé l'algorithme des k -medoids pour obtenir quatre clusters et cette algorithme retourne également quatre clusters correspondant exactement aux classes des experts. De plus les médoides (échantillons servant de représentants des clusters) calculés par cette méthode correspondent exactement aux échantillons les plus proches des points moyens obtenus par l'algorithme des k -means. Tous ces résultats confirment ce que nous avons observé sur la Figure 11.5.

Cluster	Diamètre
RUCS	23,4%
BCS	26,9%
LUCS	23,7%
Unaffected	21,0%

TABLE 11.1 – Diamètres (i.e. distances maximales entre deux chirotopes d'une même classe) des quatre classes. Ce diamètre est donné en pourcentage du nombre de bases non-fixes.

Nous allons étudier ces clusters afin de déterminer si la partition est satisfaisante ou non. Comme nous l'avons vu dans le Chapitre 5, une partition en clusters est satisfaisante si les clusters sont homogènes et s'ils sont bien différenciés. Ainsi il faut que les distances au sein d'un cluster ne soient pas trop grandes et que les distances entre les clusters ne soient pas trop petites. La Figure 11.4 nous indique que la moyenne des distances au sein d'un cluster est au maximum de 20,2%

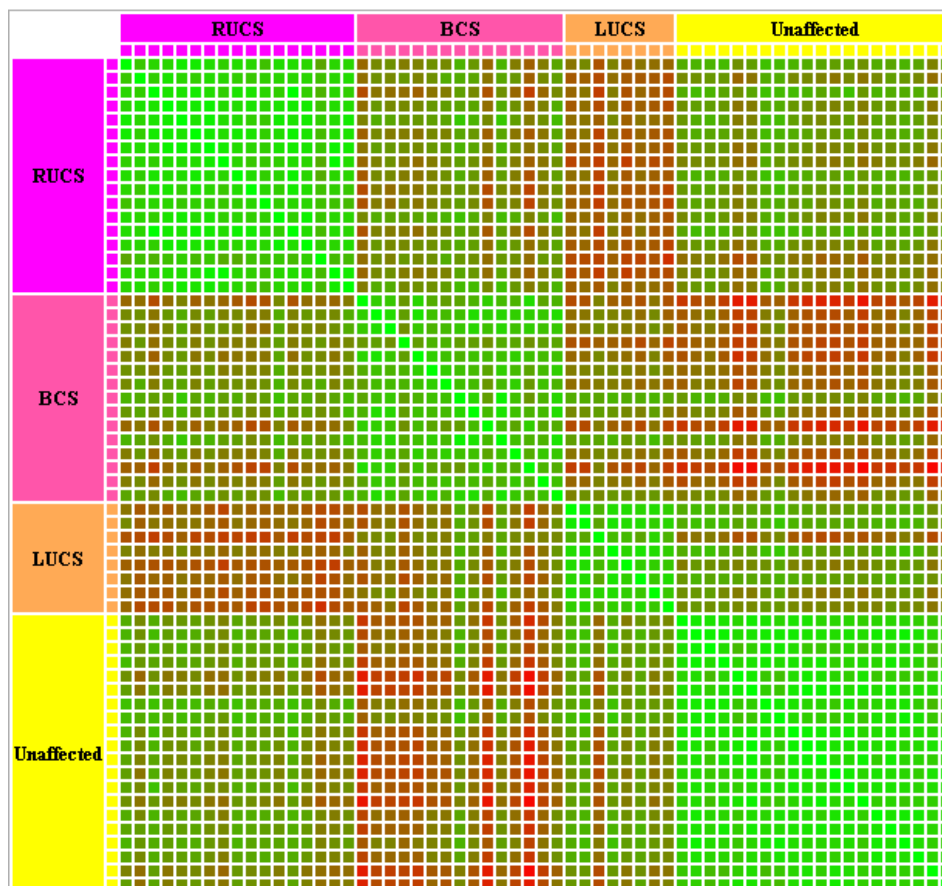


FIGURE 11.5 – Matrice des distances entre les chirotopes. L'échelle de couleur utilisée est la même que celle de la Figure 11.4.

	RUCS	BCS	LUCS
BCS	18,5%		
LUCS	28,1%	21,1%	
Unaffected	19,2%	19,9%	18,4%

TABLE 11.2 – Distances minimales entre les chirotopes des quatre classes (en pourcentage du nombre de bases non-fixes).

alors que la moyenne des distances entre cluster est au minimum de 25,9%. Ainsi il semblerait que cette partition soit satisfaisante. Pour s'en assurer nous avons calculé le diamètre de chaque classe (c'est-à-dire la distance maximale entre deux chirotopes d'un même cluster), ainsi que les distances minimales entre deux chirotopes de deux clusters différents. Ces résultats sont présentés respectivement dans les Tables 11.1 et 11.2. Les diamètres des classes sont comprises entre 21% et 26,9% et les distances minimales entre clusters sont comprises entre 18,5% et 28,1%. Ainsi les plus grandes distances au sein des clusters et les plus petites distances entre les clusters ont le même ordre de grandeur ce qui confirme que cette partition en cluster est satisfaisante.

Caractérisation des classes. Nous avons appliqué la méthode présentée dans le Chapitre 6 afin d'obtenir une caractérisation de ces classes. Nous avons alors calculé pour chaque classe \mathcal{C} et pour chaque base B , le taux de discriminabilité de B entre \mathcal{C} et $\overline{\mathcal{C}}$. Pour chaque classe nous avons trouvé plusieurs bases totalement discriminantes. Précisément, il y a 147 bases totalement discriminantes pour la classe BCS, 15 667 bases totalement discriminantes pour les LUCS, 5 064 bases totalement discriminantes pour les RUCS et 7 pour les Unaffected. Ainsi pour chaque classe, il nous suffit de regarder le signe dans un échantillon d'une base totalement discriminante pour cette classe pour savoir si cet échantillon appartient à cette classe. Nous donnerons des exemples de bases totalement discriminantes dans la section suivante.

Le nombre important de bases totalement discriminantes est cohérent avec les résultats précédents. Tous ces résultats montrent que la fusion des sutures correspond à des changements au niveau de la position relative des landmarks.

11.2 Etude en utilisant uniquement les landmarks anatomiques

Nous avons décidé de refaire une étude en n'utilisant cette fois que les 41 landmarks anatomiques. Cette étude est motivée par deux raisons : se concentrer uniquement sur ce qui a un vrai sens anatomique et réduire les temps de calculs puisque le nombre total de quadruplets de landmarks passe de $\binom{133}{4} = 12\,457\,445$ à $\binom{41}{4} = 101\,270$. Comme précédemment, les landmarks sont en position générale et chaque quadruplet de landmarks forme donc une base.

Nous avons extrait des chirotopes de l'étude précédente, les signes des bases n'utilisant que des landmarks anatomiques. Le nombre de bases fixes est cette fois-ci de 74 665 ce qui représente environ 74% du nombre total de bases. Nous n'effectuerons les calculs suivants en n'utilisant que les bases non-fixes, soit 26 605 bases.

Les résultats que nous allons aborder ici ont été présentés en partie dans [23].

Distance entre les chirotopes. La Figure 11.6 représente la moyenne des distances entre les chirotopes en fonction des classes définies par les experts. La distance la plus faible entre deux chirotopes est d'environ 12% des bases non-fixes et la distance la plus grande est d'environ 45%. Contrairement à notre étude utilisant tous les landmarks, on remarque cette fois-ci que seulement trois classes se distinguent. En effet, la moyenne des distances entre deux chirotopes de la classe BCS est égale à la moyenne des distances entre un chirotope de la classe RUCS et un chirotope de la classe Unaffected.

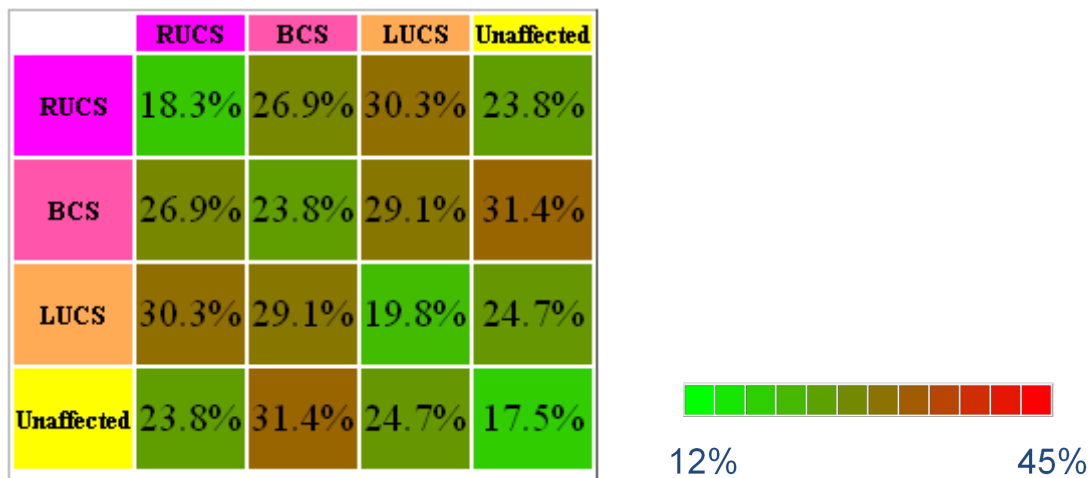


FIGURE 11.6 – Moyenne des distances entre les chirotopes en fonction des classes définies par les experts (en pourcentage du nombre de bases non-fixes).

Dans Figure 11.7, comme avec les distances moyennes, on remarque que la zone correspondant aux distances entre deux chirotopes BCS ne se démarque pas réellement du tableau.

Classification automatique. Comme dans l'étude précédente, nous avons utilisé l'algorithme des k -means pour obtenir une partition des chirotopes en quatre clusters. Les clusters fournis par l'algorithme correspondent aux classes des experts à une exception près : le crâne étiqueté "600000". Ce crâne a les deux cotés de la suture fusionnés (il appartient donc à la classe des BCS) mais il est placé dans la partition contenant tous les crânes LUCS.

Ce crâne est très particulier puisque si on lance l'algorithme des k -means avec les classes des experts comme initialisation des clusters, alors les clusters correspondent aux classes des experts sauf pour le crâne "600000". Ce crâne appartient cette fois-ci au cluster contenant tous les RUCS.

Nous avons également utilisé l'algorithme des k -medoids afin de partitionner l'ensemble des chirotopes en quatre clusters. Cette fois il y a deux changements par rapport aux classes des experts. En plus du crâne "600000" un autre crâne BCS (étiqueté "600016") appartient lui aussi au cluster contenant tous les crânes RUCS.

Nous avons étudié ces quatre partitionnements afin de déterminer s'ils créent des clusters homogènes et bien différenciés. Les diamètres des clusters pour les différentes classifications sont répertoriés dans la Table 11.3. Quelle que soit la classification, les diamètres des clusters sont compris entre 24,6% et 33,5%. On remarque que les diamètres des clusters sont identiques dans les différentes classifications, sauf pour le cluster de la classification provenant des k -medoids contenant les chirotopes des RUCS et des crânes "600000" et "600016". On peut en déduire que le crâne "600000" n'intervient pas dans les calculs des distances maximales au sein des clusters. Par contre, dans la classification provenant des k -medoids, le crâne "600016" intervient dans le calcul du diamètre. Ce diamètre étant plus grand que dans les autres classifications, la classification provenant des k -medoids semble moins satisfaisante que les autres.

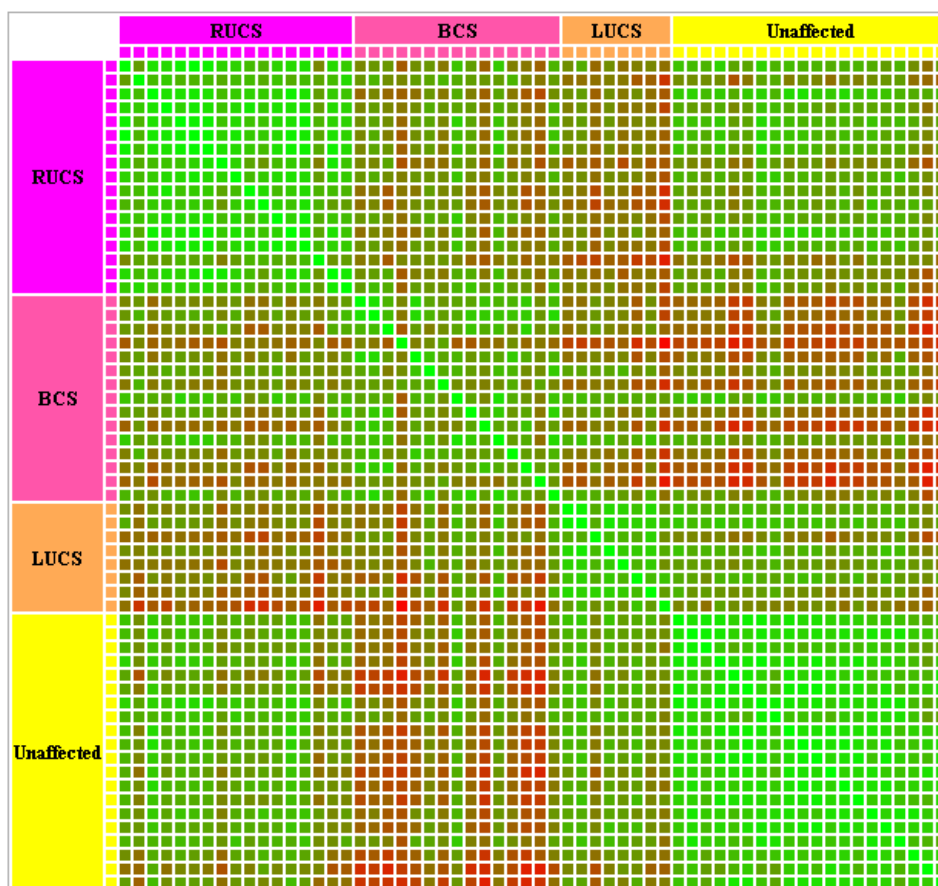


FIGURE 11.7 – Matrice des distances entre les chirotopes. L'échelle de couleur utilisée est la même que celle de la Figure 11.6.

La Table 11.4 répertorie pour chaque classification les distances minimales entre deux clusters. Cette fois-ci, les crânes "600000" et "600016" interviennent dans les calculs de ces distances puisque les valeurs sont différentes d'une classification à une autre. Globalement, ces distances minimales sont comprises entre 15,7% et 22,9%. En conclusion, les distances minimales entre les clusters sont plus petites que les diamètres des clusters.

Tous ces résultats nous confirment ce que nous avons remarqué avec les Figures 11.6 et 11.7, c'est-à-dire que contrairement à notre étude portant sur tous les landmarks, il n'y a pas de classification qui apparaît clairement.

Caractérisation des classes. Bien que lorsque l'on réalise une classification automatique des données on n'obtient pas exactement les classes des experts, nous avons cherché s'il est possible de trouver des sous-ensembles de bases permettant de discriminer les classes des experts. Nous avons trouvé 22 bases totalement discriminantes pour les LUCS et 4 bases totalement discriminantes pour les RUCS. Les quatre bases qui sont totalement discriminantes pour les RUCS contiennent toutes

Classe	Diamètre	Cluster	Diamètre
RUCS	28,6%	RUCS	28,6%
BCS	33,5%	BCS - "600000"	33,5%
LUCS	26,8%	LUCS + "600000"	26,8%
Unaffected	24,6%	Unaffected	24,6%

Cluster	Diamètre	Cluster	Diamètre
RUCS + "600000"	28,6%	RUCS + {"600000", "600016"}	29,4%
BCS - "600000"	33,5%	BCS - {"600000", "600016"}	33,5%
LUCS	26,8%	LUCS	26,8%
Unaffected	24,6%	Unaffected	24,6%

TABLE 11.3 – Diamètres (i.e. distances maximales entre deux chirotopes d’une même classe) des clusters pour les différentes classifications. Il s’agit des classes des experts (en haut à gauche), des meilleurs clusters obtenus par l’algorithme des k -means (en haut à droite), des clusters obtenus par l’algorithme des k -means en initialisant avec les classes des experts (en bas à gauche) et des meilleurs clusters obtenus par l’algorithme des k -medoids (en bas à droite). Les diamètres sont donnés en pourcentage du nombre de bases non-fixes.

deux landmarks anatomiques : le *bregma* (qui est situé vers le sommet du crâne et correspond à l’intersection des sutures coronale et sagittale) et le *lambda* (situé vers l’arrière du crâne et correspondant à l’intersection des sutures sagittales et lambdoïde). Chacune de ces bases contient un troisième point situé sur le plan médian sagittal. Ce troisième landmark est le *nasion* (situé entre les yeux, juste au-dessus de l’arête du nez) pour deux de ces bases et le *point nasal* (racine du nez) pour les deux autres bases. Pour chaque base, le quatrième landmarks est un point de la partie droite de la base du crâne (dans la zone entre le palais et l’orbite de l’œil droit). A noter que pour chacune de ces bases, si l’on remplace le quatrième landmark par son symétrique par rapport au plan médian sagittal, on obtient l’une des 22 bases totalement discriminantes pour les RUCS. La Figure 11.8 représente une des quatre bases totalement discriminante pour les RUCS et la base symétrique qui est totalement discriminante pour les LUCS.

Pour les BCS, il n’y a pas de bases totalement discriminantes. Cependant, nous avons trouvé deux bases B_1 et B_2 telles qu’un chirotope appartient à la classe BCS si et seulement si le signe de ces deux bases est égal à $-$ (en notant χ_M ce chirotope, on a alors $\chi_M(B_1) = \chi_M(B_2) = -$).

En reprenant la terminologie de la Section 6.1, cela signifie que nous avons caractérisé cette classe comme étant égale à l’ensemble des chirotopes $\mathcal{C}_{\mathcal{D}_1, x, r}$ où $\mathcal{D}_1 = \{B_1, B_2\}$, $r = 0$ et x est un point de $\{-1, 1\}^{\mathcal{B}}$ (où \mathcal{B} est l’ensemble des bases non-fixes pour ces données) tel que $x(B_1) = x(B_2) = -1$.

On rappelle qu’en représentant les chirotopes par des points de $\mathbb{R}^{\mathcal{B}}$, l’ensemble $\mathcal{C}_{\mathcal{D}_1, x, r}$ est l’ensemble des chirotopes contenus dans une boule de $\mathbb{R}^{\mathcal{B}}$ centrée en x et de rayon r pour la distance $d_{\mathcal{D}}$.

Bien que n’étant pas totalement discriminantes, ces deux bases ont un taux de discriminabilité élevé puisque l’on a $\tau(B_1, \mathcal{C}, \overline{\mathcal{C}}) = 0,91$ et $\tau(B_2, \mathcal{C}, \overline{\mathcal{C}}) = 0,89$ où \mathcal{C} est la classe BCS et $\overline{\mathcal{C}}$ est son complémentaire. Ces deux bases ont deux landmarks en commun. Nous allons noter $B_1 = \{1234\}$

	RUCS	BCS	LUCS
BCS	17,7%		
LUCS	22,9%	19,6%	
Unaffected	15,7%	18,2%	18,1%

	RUCS	BCS - "600000"	LUCS + "600000"
BCS - "600000"	19,6%		
LUCS + "600000"	17,7%	16,8%	
Unaffected	15,7%	21,6%	18,1%

	RUCS + "600000"	BCS - "600000"	LUCS
BCS - "600000"	15,9%		
LUCS	20,7%	19,6%	
Unaffected	15,7%	21,6%	18,1%

	RUCS + {"600000", "600016"}	BCS - {"600000", "600016"}	LUCS
BCS - {"600000", "600016"}	16,8%		
LUCS	19,6%	21,6%	
Unaffected	15,7%	21,6%	18,1%

TABLE 11.4 – Distances minimales (en pourcentage du nombre de bases non-fixes) entre les chirotopes des clusters pour les différentes classifications. De haut en bas, ces classifications sont : classes des experts, meilleurs clusters obtenus par l’algorithme des k -means, clusters obtenus par l’algorithme des k -means en initialisant avec les classes des experts et meilleurs clusters obtenus par l’algorithme des k -medoids.

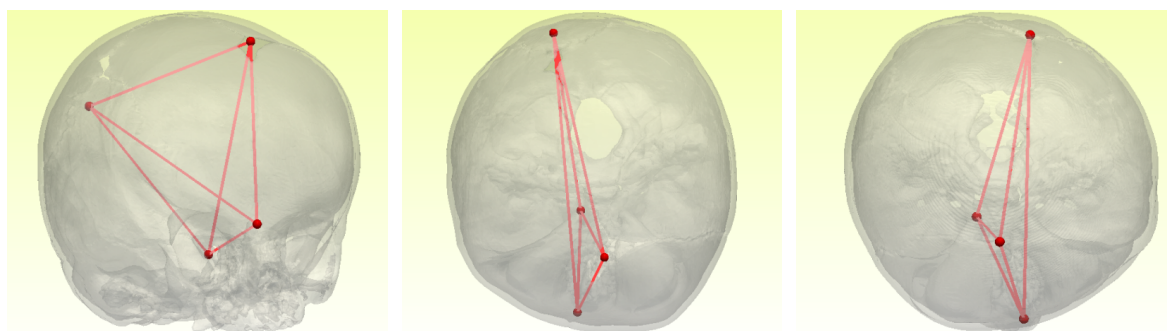


FIGURE 11.8 – Une des quatre bases totalement discriminantes pour les RUCS (à gauche et au milieu) et la base obtenue par symétrie qui est totalement discriminante pour les LUCS (à droite).

et $B_2 = \{1256\}$. La Figure 11.9 représente la position des bases B_1 et B_2 dans un crâne de BCS. La Figure 11.10 détaille la position de ces bases pour la classe BCS et la Figure 11.11 représente les autres possibilités qui correspondent à des crânes n'appartenant pas à la classe BCS.

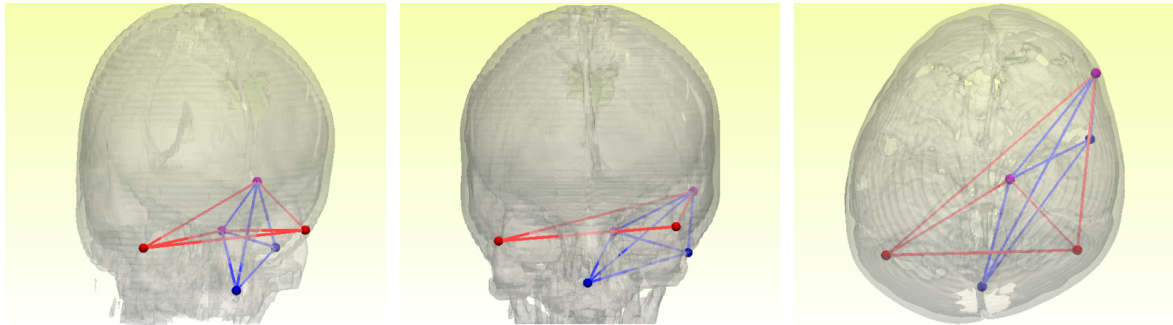


FIGURE 11.9 – Représentation des deux bases (B_1 en rouge et B_2 en bleu) nous donnant une caractérisation des crânes BCS.



FIGURE 11.10 – Représentation des bases B_1 (rouge) et B_2 (bleu) telle que l'orientation des deux bases soit $-$, ce qui correspond à une caractérisation des crânes BCS.

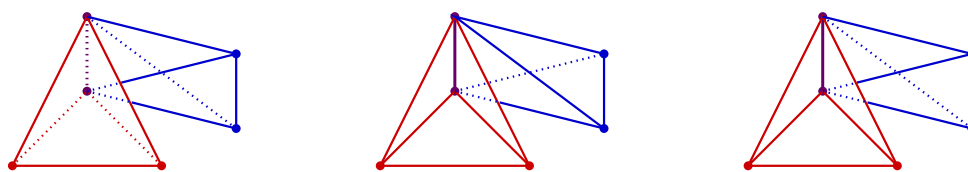


FIGURE 11.11 – Les autres orientations possibles $[-, +]$, $[+, -]$ et $[+, +]$ (de gauche à droite), correspondant à des crânes n'appartenant pas à la classe BCS.

Les caractérisations des classes que nous venons de voir nous donnent une caractérisation indirecte de la classe Unaffected. En effet, un chirotope appartient à la classe Unaffected s'il n'appartient à aucune des trois autres classes. Nous avons tout de même cherché s'il existe une caractérisation directe de la classe Unaffected qui utiliserait un sous-ensemble de bases. Comme il n'existe pas de bases totalement discriminantes pour la classe Unaffected, nous avons essayé d'obtenir une caractérisation forte de cette classe (voir Section 6.2.2).

Nous avons trouvé un sous-ensemble \mathcal{D}_2 de cinq bases (dont les taux de discriminabilité sont compris entre 0,85 et 0,875) et deux points y et \bar{y} de $\{-1, 1\}^{\mathcal{B}}$ nous donnant une caractérisation forte de la classe Unaffected. En notant \mathcal{C} la classe Unaffected et $\overline{\mathcal{C}}$ son complémentaire, on a alors $\mathcal{C} = \mathcal{C}_{\mathcal{D}_2, y, 2}$ et $\overline{\mathcal{C}} = \mathcal{C}_{\mathcal{D}_2, \bar{y}, 2}$. Ainsi, un chirotope appartient à la classe Unaffected si et seulement si il existe au moins trois bases de \mathcal{D}_2 telles que pour chacune de ces bases, le signe de la base dans le chirotope soit égal au signe de la coordonnée de y pour cette base. De même, un chirotope appartient au complémentaire de la classe Unaffected si et seulement si il existe au moins trois bases de \mathcal{D}_2 telles que pour chacune de ces bases, le signe de la base dans le chirotope soit égal au signe de la coordonnée de \bar{y} pour cette base.

En conclusion, les changements de positions relatives des landmarks anatomiques permettent de discriminer les crânes non-affectés des crânes présentant une craniosynostose coronale et permet également de caractériser les types de craniosynostose coronale (LUCS, RUCS, BCS).

11.3 Etude de la symétrie des crânes

Contexte

Lors de leur étude, Heuzé et al. [31] ont abordé la question de l'asymétrie des déformations de crânes. Plus précisément, les auteurs étudient trois hypothèses. La première hypothèse (H1) est que lorsque la fusion prématurée de la suture a lieu à la fois sur la partie gauche et la partie droite du crâne (i.e. crânes appartenant à la classe BCS), alors la déformation affecte les deux côtés de façon similaire et la symétrie bilatérale des crânes est donc conservée. La deuxième hypothèse (H2) est que les classes LUCS et RUCS sont des images miroirs l'une de l'autre. Cette deuxième hypothèse signifierait qu'il n'y a qu'un seul processus de déformation lorsque la fusion prématurée de la suture n'a lieu que d'un seul côté. La troisième hypothèse (H3) est que le processus de déformation est le même sur les côtés des crânes présentant une fusion de la suture (les deux côtés pour les BCS, le côté gauche des LUCS et le côté droit des RUCS).

On peut partitionner les landmarks et semi-landmarks en trois parties : 14 landmarks et 35 semi-landmarks sont sur le côté gauche du crâne, 14 landmarks et 35 semi-landmarks sont sur le côté droit du crâne et 13 landmarks et 22 semi-landmarks sont sur le plan médian sagittal. Comme les landmarks et semi-landmarks des côtés gauches et droits sont en correspondance, les auteurs ont défini des *demi-crânes* comme étant l'ensemble des landmarks et semi-landmarks d'un côté auxquels ils ont rajouté les landmarks et semi-landmarks situés sur le plan médian sagittal. Ainsi chaque demi-crâne est composé de 27 landmarks et 57 semi-landmarks. Pour pouvoir comparer les demi-crânes, une image miroir des demi-crânes est alors calculée et les landmarks des images miroirs ont été ré-étiquetés en utilisant la numérotation des demi-crânes originels.

Pour plus de compréhension concernant les résultats, nous parlerons de demi-crânes gauches et demi-crânes droits même si lors des calculs ce sont en fait 60 demi-crânes gauches et 60 images miroirs de demi-crânes droits qui sont utilisées. L'étude réalisée par Heuzé et al. et notre étude utilisent donc huit classes de demi-crânes : demi-crânes gauche et droit des Unaffected, BCS, LUCS et RUCS. Nous noterons respectivement L-Unaffected, R-Unaffected, L-BCS, R-BCS, L-LUCS, R-LUCS, L-RUCS et R-RUCS ces classes. Le but est d'étudier les hypothèses (H1), (H2) et (H3) en cherchant

quelles sont les classes de demi-crânes les plus proches et quelles sont les classes les plus éloignées parmi ces huit classes.

Résultats de Heuzé et al. [31]. Les résultats de l'Analyse en Composantes Principales réalisée par Heuzé et al. [31] sur les 120 demi-crânes sont données par la Figure 11.12. Les deux premiers axes de cette ACP permettent de séparer ces 120 demi-crânes en 4 groupes : les demi-crânes gauches et droits des individus non-affectés dans le quart supérieur gauche du graphique, les demi-crânes gauches et droits des BCS dans le quart inférieur droit, les demi-crânes des RUCS et LUCS ne présentant pas de fusion de la suture (demi-crânes gauche des RUCS et demi-crânes droits des LUCS) dans le quart inférieur gauche, et les demi-crânes des RUCS et LUCS présentant une fusion de la suture (demi-crânes droits des RUCS et demi-crânes gauches des LUCS) dans le quart supérieur droit du graphique.

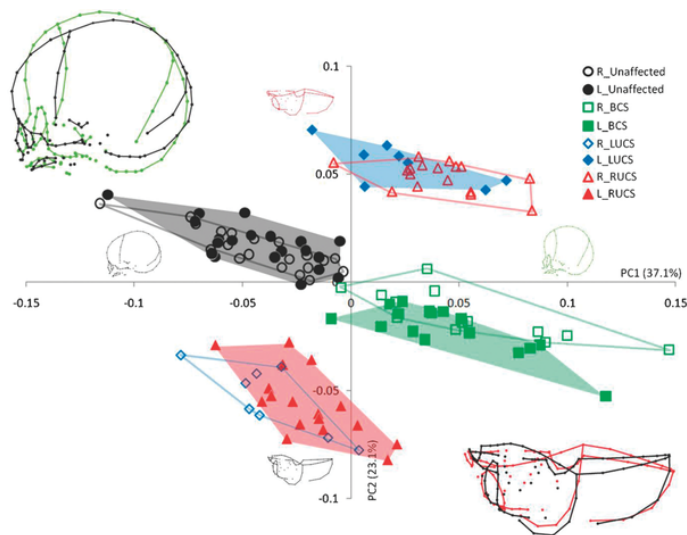


FIGURE 11.12 – Position des 120 demi-crânes sur les deux premiers axes de l'Analyse en Composantes Principales réalisée par Heuzé et al. [31].

Heuzé et al. ont également calculé la distance de Procrustes entre les huit classes de demi-crânes. Pour toute paire de classes de demi-crânes, la Table 11.5 indique la distance de Procrustes entre les formes moyennes de ces deux classes. Le chiffre entre parenthèse est le résultat d'un test de permutation dans lequel 10 000 répartitions aléatoires ont été réalisées pour chaque paire de classes. Plus cette proportion est petite et plus les classes \mathcal{C}_1 et \mathcal{C}_2 sont séparées.

Cette table montre que parmi toutes les paires de classes de demi-crânes seulement quatre paires de classes ne sont pas bien séparées : les parties gauches et droites des crânes des individus non-affectés ($\{R\text{-Unaffected}, L\text{-Unaffected}\}$), les demi-crânes gauches et droits des BCS ($\{R\text{-BCS}, L\text{-BCS}\}$), les demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture ($\{R\text{-LUCS}, L\text{-RUCS}\}$) et les demi-crânes des LUCS et RUCS dont la suture est fusionnée ($\{R\text{-RUCS}, L\text{-LUCS}\}$).

Mise en œuvre de notre méthode

Nous avons également réalisé une étude des symétries et asymétries des crânes. Pour cette étude, il nous fallait les chirotopes associés aux demi-crânes. Nous avons choisi de garder tels quels les demi-crânes gauches et de symétriser les demi-crânes droits. Pour le calcul des demi-crânes, nous pouvions soit calculer le chirotope associé aux 84 landmarks de chaque demi-crâne ou extraire ce chirotope du chirotope associés au crâne tout entier. Nous avons opté pour la deuxième solution pour ne pas avoir à faire plusieurs fois les mêmes calculs.

Nous avons partitionné les bases d'un matroïde orienté en trois groupes : les bases utilisant uniquement des landmarks du demi-crâne gauche, les bases utilisant uniquement des landmarks du demi-crâne droit et les bases utilisant à la fois des landmarks du demi-crâne gauche et du demi-crâne droit. Ainsi pour chaque matroïde orienté, le premier groupe de bases forme le matroïde orienté associé au demi-crâne gauche et le deuxième groupe forme le matroïde orienté associé au demi-crâne droit.

A partir de ce deuxième groupe nous avons calculé le chirotope associé à l'image miroir du demi-crâne droit. Ce calcul consiste essentiellement à inverser le signe des bases. Or à chaque tétraèdre de l'image miroir du demi-crâne droit correspond un tétraèdre du demi-crâne gauche. Nous avons alors réordonné les bases des matroïdes orientés associés aux images miroirs pour pouvoir comparer plus facilement les bases qui se correspondent. La Section 8.3 fournit plus de détails sur le calcul des demi-crânes. Nous n'utiliserons pas le troisième groupe de bases dans cette étude des symétries et asymétries.

Distance entre les classes. Pour analyser les demi-crânes, nous avons commencé par étudier les chirotopes associés aux demi-crânes. Chaque demi-crâne étant composé de 84 landmarks, chaque chirotope associé à un demi-crâne est donc composé de $\binom{84}{4} = 1\,929\,501$ bases. Or parmi toutes ces bases nous avons remarqués qu'il n'y en a que 548 528 non-fixes. Pour savoir si cette distance peut permettre de séparer ou de regrouper des classes de demi-crânes, nous avons commencé par étudier la moyenne des distances entre les matroïdes orientés de deux classes. Ces données sont représentées dans la Table 11.6.

On observe alors que les distances moyennes les plus faibles sont sur la diagonale (i.e. lorsque l'on calcule la distance moyenne entre deux chirotopes d'une même classe) et pour les quatre paires de classes exhibées par Heuzé et al. : les parties gauches et droites des crânes des individus non-affectés ({R-Unaffected, L-Unaffected}), les demi-crânes gauches et droits des BCS ({R-BCS, L-BCS}), les demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture ({R-LUCS, L-RUCS}) et les demi-crânes des LUCS et RUCS dont la suture est fusionnée ({R-RUCS, L-LUCS}). En effet il n'y a que sur la diagonale et sur ces quatre blocs que la moyenne est inférieure à 21% alors qu'elle est supérieur à 25,5% pour les autres blocs.

Comme les données de la Table 11.6 ne sont que des moyennes de distances, nous avons voulu regarder de plus près la distance entre les matroïdes orientés. La Figure 11.13 représente le tableau des distances entre toutes les paires de matroïdes orientés associés à un demi-crâne sous forme d'un dégradé de couleur : plus la couleur est verte et plus les matroïdes orientés sont proches et à l'inverse plus la couleur est rouge est plus les matroïdes orientés sont différents. Les valeurs dans

	L-BCS	L-Unaffected	L-LUCS	L-RUCS	R-BCS	R-Unaffected	R-LUCS
L-Unaffected	0,0978 (0,0001)						
L-LUCS	0,0829 (0,0001)	0,0803 (0,0001)					
L-RUCS	0,0755 (0,0001)	0,0787 (0,0001)	0,1214 (0,0001)				
R-BCS	0,0207 (0,7686)	0,1016 (0,0001)	0,0792 (0,0001)	0,0848 (0,0001)			
R-Unaffected	0,0966 (0,0001)	0,0127 (0,8737)	0,0813 (0,0001)	0,0765 (0,0001)	0,0989 (0,0001)		
R-LUCS	0,0928 (0,0001)	0,0768 (0,0001)	0,1269 (0,0001)	0,0303 (0,0759)	0,0995 (0,0001)	0,073 (0,0001)	
R-RUCS	0,0743 (0,0001)	0,0915 (0,0001)	0,0283 (0,1249)	0,1221 (0,0001)	0,0672 (0,0001)	0,0909 (0,0001)	0,1308 (0,0001)

TABLE 11.5 – Distance de Procrustes et résultats du test de permutation sur les huit classes de demi-crânes de demi-crânes d'après Heuzé et al. [31]. Les valeurs en gras indiquent que les deux classes de demi-crânes sont proches.

	L-BCS	L-Unaffected	L-LUCS	L-RUCS	R-BCS	R-Unaffected	R-LUCS	R-RUCS
L-BCS	19,9%							
L-Unaffected	28,3%	16,5%						
L-LUCS	29,2%	27,0%	14,7%					
L-RUCS	27,9%	26,0%	40,0%	14,4%				
R-BCS	20,7%	28,8%	26,5%	31,4%	19,8%			
R-Unaffected	28,4%	17,1%	27,5%	25,6%	28,3%	16,1%		
R-LUCS	28,6%	26,3%	40,4%	15,4%	31,5%	25,1%	15,0%	
R-RUCS	29,0%	27,1%	14,9%	39,6%	25,6%	27,1%	39,6%	14,4%

TABLE 11.6 – Distance moyenne entre les différentes classes de demi-crânes (en pourcentage du nombre total de bases non-fixes). Les valeurs en gras indiquent que les deux classes de demi-crânes sont proches.

ce tableau varie d'environ 10% (vert clair) à 46% (rouge). Le tableau étant symétrique par rapport à la diagonale, nous allons nous concentrer uniquement sur la partie inférieure du tableau.

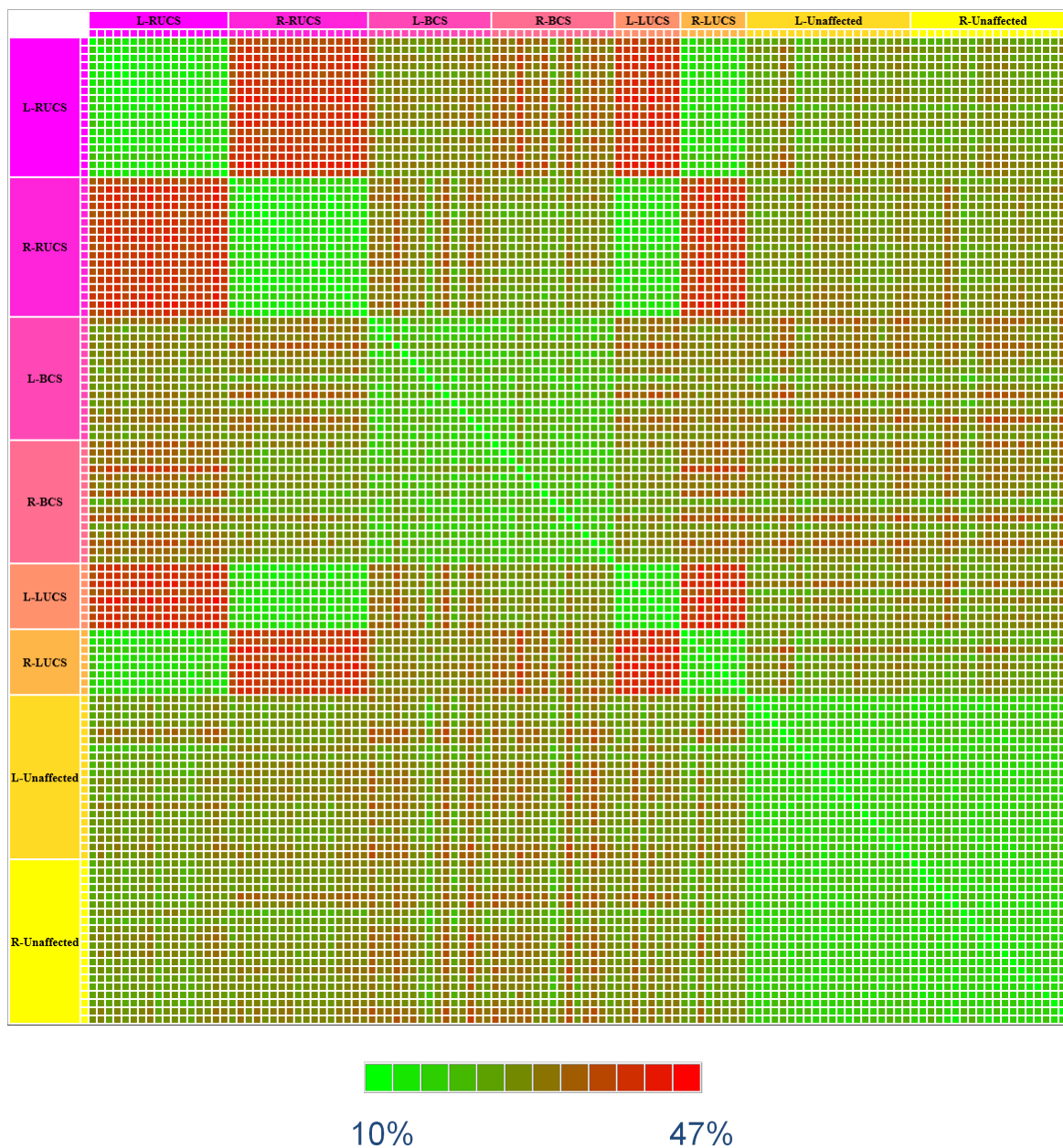


FIGURE 11.13 – Tableau des distances entre toutes les paires de demi-crânes.

Dans ce tableau on observe que la diagonale est formée de blocs de couleur vert ce qui signifie que quelle que soit la classe de demi-crâne que l'on considère, alors deux matroïdes orientés associés à deux demi-crânes de cette classe ont peu de différences. On remarque également que quatre

autres blocs verts ressortent. Trois de ces blocs sont vert clair, il s'agit des blocs correspondant aux couples {R-Unaffected, L-Unaffected}, {R-RUCS, L-LUCS} et {R-LUCS, L-RUCS}. Le quatrième bloc est légèrement plus foncé et correspond au couple {R-BCS, L-BCS}.

Classification automatique. Pour confirmer l'impression visuelle donnée par la Figure 11.13 nous avons cherché à obtenir quatre clusters par l'algorithme des k -means. Comme les matroïdes orientés associés aux demi-crânes sont tous uniformes, nous représentons chaque chirotope associé à un demi-crâne par un point de $\{-1, 1\}^b$ où b est le nombre de bases non-fixes, soit 548 528 ici (c.f. Section 5.1.2). Le résultat est alors ce que nous observons sur la Figure 11.13 : le premier cluster est formé par les demi-crânes R-BCS et L-BCS, le deuxième cluster est formé par les demi-crânes R-Unaffected et L-Unaffected, le troisième cluster est formé par les demi-crânes R-RUCS et L-LUCS et le dernier cluster est formé par les demi-crânes R-LUCS et L-RUCS.

Cluster	Diamètre
L-RUCS et R-LUCS	21,3%
L-LUCS et R-RUCS	20,9%
L-BCS et R-BCS	29,9%
L-Unaffected et R-Unaffected	22,2%

TABLE 11.7 – Diamètre des quatre classes obtenues par l'algorithme des k -means (i.e. distance maximale entre deux matroïdes orientés d'une même classe). Ce diamètre est donné en pourcentage du nombre de bases non-fixes.

	L-RUCS et R-LUCS	L-LUCS et R-RUCS	L-BCS et R-BCS
L-LUCS et R-RUCS	32,7%		
L-BCS et R-BCS	20,1%	17,3%	
L-Unaffected et R-Unaffected	17,6%	20,0%	17,5%

TABLE 11.8 – Distances minimales entre les chirotopes des quatre classes obtenues par l'algorithme des k -means (en pourcentage du nombre de bases non-fixes).

A ce stade, on peut se poser la question de savoir si cette partition est satisfaisante ou non. Nous avons calculé dans un premier temps le diamètre de chaque cluster c'est-à-dire la distance maximale entre deux matroïdes orientés associés à deux demi-crânes d'un même cluster. Les résultats sont représentés dans la Table 11.7.

Nous avons ensuite calculé la distance entre les différents clusters. Nous avons regardé dans un premier temps les distances minimales entre ces quatre clusters. Ces valeurs sont reportées dans la Table 11.8. On remarque que deux clusters sont très séparés : il s'agit du cluster formé des demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture (L-RUCS et R-LUCS) et les demi-crânes des LUCS et RUCS dont la suture est fusionnée (L-LUCS et R-RUCS). On remarque également que les distances minimales entre le cluster formé des demi-crânes des Unaffected (L-Unaffected et R-Unaffected) et l'un des deux clusters précédents sont légèrement inférieures aux diamètres de ces

clusters. Ainsi il semble que le cluster des Unaffected soit séparé des clusters {L-RUCS et R-LUCS} et {L-LUCS et R-RUCS}. Par contre le cluster composé des demi-crânes des BCS (L-BCS et R-BCS) ne semble pas aussi bien séparé des autres clusters. En effet, le diamètre du cluster des BCS est bien plus grand que les distances minimales entre un chirotope des BCS et un chirotope des autres clusters.

Cluster	Distance maximale
L-RUCS et R-LUCS	189 808,2
L-LUCS et R-RUCS	182 249,9
L-BCS et R-BCS	266 091,4
L-Unaffected et R-Unaffected	196 407,6

TABLE 11.9 – Distance maximale (distance L_1 dans $\{-1, 1\}^b$ où b est le nombre de bases non-fixes) pour chaque cluster entre le point moyen du cluster et un chirotope associé à un demi-crâne du cluster.

		Chirotope			
		L-RUCS et R-LUCS	L-LUCS et R-RUCS	L-BCS et R-BCS	L-Unaffected et R-Unaffected
Point moyen	L-RUCS et R-LUCS		397 496,1	258 042,9	221 824,3
	L-LUCS et R-RUCS	399 194,4		233 042,7	240 691,6
	L-BCS et R-BCS	282 978,8	266 888,2		265 427,9
	L-Unaffected et R-Unaffected	247 127,7	255 867,4	239 882,1	

TABLE 11.10 – Distance minimale (distance L_1 dans $\{-1, 1\}^b$ où b est le nombre de bases non-fixes) entre le point moyen d'un cluster et un chirotope associé à un demi-crâne d'un autre cluster.

Pour confirmer que les clusters sont séparés, nous pouvons également utiliser les points moyens obtenus lors de l'exécution de l'algorithme des k -means. La Table 11.9 indique la distance maximale pour chaque cluster entre le point moyen du cluster et un chirotope associé à un demi-crâne du cluster. Pour réaliser ces calculs nous avons utilisé la distance L_1 dans l'espace $\{-1, 1\}^b$ où b est le nombre de bases non-fixes. On remarque alors que le cluster des BCS est le cluster possédant des chirotopes les plus éloignés du point moyen du cluster.

Nous avons ensuite calculé pour toute paire de clusters, les distances minimales entre le point moyen du premier cluster et les chirotopes du deuxième cluster (Table 11.10). Tout comme pour la Table 11.8, on remarque que deux clusters sont très séparés : il s'agit du cluster formé des demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture (L-RUCS et R-LUCS) et les demi-crânes des LUCS et RUCS dont la suture est fusionnée (L-LUCS et R-RUCS). En effet, la distance

minimale entre le point moyen d'un de ces deux clusters et un chirotope de l'autre cluster est plus de deux fois plus grande que la distance maximale entre ce point moyen et un chirotope du même cluster. Le cluster formé des demi-crânes Unaffected (L-Unaffected et R-Unaffected) est également bien séparé des deux clusters précédents. En effet, les distances de la Table 11.10 entre les Unaffected et les deux clusters précédents ($\{L\text{-RUCS}, R\text{-LUCS}\}$ et $\{L\text{-LUCS}, R\text{-RUCS}\}$) sont plus grandes que les distances de la Table 11.9 pour ces trois clusters.

En étudiant d'un peu plus près les crânes qui sont utilisés lors du calcul des distances minimales pour le cluster BCS (que ce soit les distances minimales entre deux chirotopes ou entre un chirotope d'un BCS et le point moyen d'un autre cluster), on se rend compte qu'un crâne est utilisé pour calculer toutes ces distances minimales. Ce crâne est celui de l'individu "600000" qui était classé précédemment comme RUCS au lieu de BCS (c.f. Section 11.2). Plus précisément, le chirotope le plus proche d'un chirotope du cluster $\{L\text{-RUCS}, R\text{-LUCS}\}$ est le même que le chirotope le plus proche du point moyen de ce cluster $\{L\text{-RUCS}, R\text{-LUCS}\}$ et correspond au demi-crâne gauche de l'individu "600000". Le chirotope le plus proche d'un chirotope du cluster $\{L\text{-LUCS}, R\text{-RUCS}\}$ est le même que le chirotope le plus proche du point moyen de ce cluster $\{L\text{-LUCS}, R\text{-RUCS}\}$ et correspond au demi-crâne droit de l'individu "600000". Par contre, le chirotope le plus proche d'un chirotope du cluster Unaffected correspond au demi-crâne droit de l'individu "600000" et le chirotope le plus proche du point moyen du cluster Unaffected correspond au demi-crâne gauche de l'individu "600000".

Ce crâne a également une autre spécificité puisqu'il s'agit du crâne le plus symétrique des crânes BCS. En effet, la distance entre le demi-crâne gauche et le demi-crâne droit de l'individu "600000" est de 15,5% des bases non-fixes alors que la distance entre la partie gauche et la partie droite d'un autre crâne BCS est comprise entre 16,1% et 27,8%.

Ainsi les chirotopes correspondants aux demi-crâne gauche et le demi-crâne droit de l'individu "600000" semble être atypique et expliquerait pourquoi le cluster BCS (L-BCS et R-BCS) est si proche des autres clusters dans les Tables 11.8 et 11.10.

Caractérisation des classes. Nous avons ensuite cherché pour chaque cluster \mathcal{C} , des bases nous permettant de discriminer \mathcal{C} des autres clusters. Pour chaque cluster \mathcal{C} , on notera $\overline{\mathcal{C}}$ l'ensemble des demi-crânes n'appartenant pas à \mathcal{C} . Ainsi pour chaque cluster \mathcal{C} et pour chaque base B , nous avons calculé le taux de discriminabilité de B entre \mathcal{C} et $\overline{\mathcal{C}}$ ($\tau(B, \mathcal{C}, \overline{\mathcal{C}})$).

Pour le cluster constitué des demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture (L-RUCS et R-LUCS), nous avons trouvé 1399 bases dont le taux de discriminabilité est égal à 1. Cela signifie que si l'on prend une seule de ces 1399 bases, alors en ne regardant que le signe de cette base dans un matroïde orienté M on peut déterminer si M est associé à un demi-crâne de LUCS ou de RUCS ne présentant pas de fusion de la suture.

Pour le cluster constitué des demi-crânes des LUCS et RUCS présentant une fusion de la suture (L-LUCS et R-RUCS), nous avons trouvé 1819 bases dont le taux de discriminabilité est égal à 1.

Pour le cluster constitué des demi-crânes des BCS (L-BCS et R-BCS), nous avons trouvé 4 bases dont le taux de discriminabilité est égal à 1. Ces quatre bases n'utilisent que des semi-landmarks. La Figure 11.14 indique la position des semi-landmarks pour deux de ces bases. Chacune de ces bases est composée de trois semi-landmarks situés sur la face du crâne et d'un sur l'avant de la voûte crânienne. La Figure 11.15 indique le changement de la position relative des landmarks pour l'une

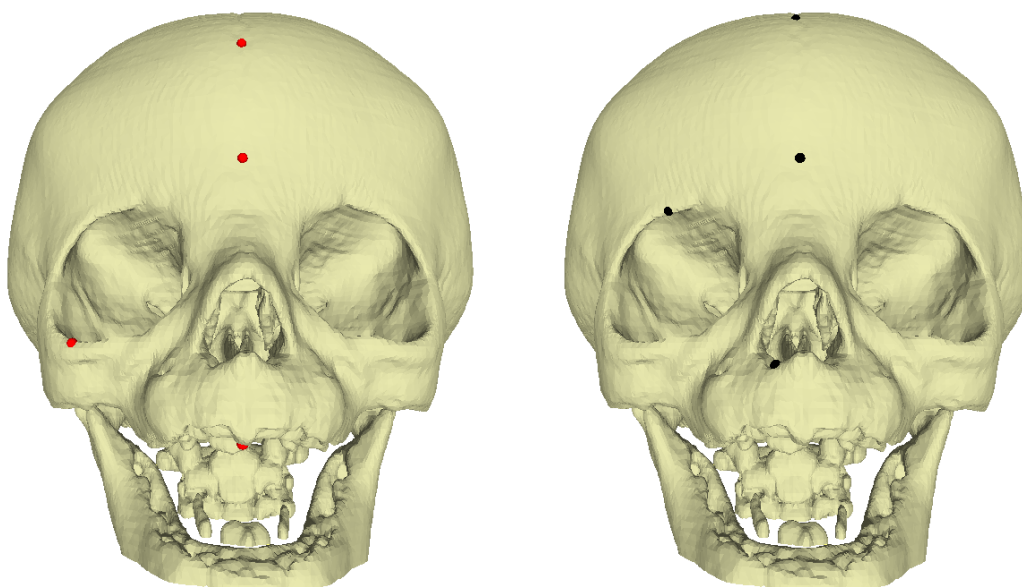


FIGURE 11.14 – Deux exemples de quatre landmarks formant une base totalement discriminante entre le cluster {L-BCS, R-BCS} et les trois autres clusters.

de ces quatre bases. On observe alors que pour un crâne BCS, le semi-landmarks situé sur l'avant de la voûte crânienne est en avant par rapport au plan formé par les trois autres semi-landmarks. Ce même semi-landmarks est en arrière du plan pour les autres crânes. On remarque sur les deux images de droite que l'angle entre les deux plans (le plan formé par les trois semi-landmarks de la face et le plan formé des deux oreilles internes et du semi-landmark de l'orbite droite) est le même pour les deux crânes. Ainsi le changement d'orientation n'est pas dû à une inclinaison du plan mais au passage du semi-landmark de la voûte crânienne de l'autre côté du plan. Ce résultat semble indiquer que contrairement aux autres crânes, la croissance des crânes BCS sur l'axe antéro-postérieur est plus important au niveau de la voûte crânienne que sur le reste du crâne.

Par contre pour le cluster constitué des demi-crânes des Unaffected (L-Unaffected et R-Unaffected), il n'existe pas de bases dont le taux de discriminabilité est égal à 1. On note \mathcal{C} ce cluster (i.e. $\mathcal{C} = \text{L-Unaffected} \cup \text{R-Unaffected}$) et $\overline{\mathcal{C}}$ l'ensemble des demi-crânes n'appartenant pas à \mathcal{C} . L'une des caractérisations que nous avons trouvés consiste à utiliser trois bases dont les taux de discriminabilité entre \mathcal{C} et $\overline{\mathcal{C}}$ sont compris entre 0,969 et 0,975. On note \mathcal{D} l'ensemble de ces trois bases. Cette caractérisation est qu'un matroïde orienté M associé à un demi-crâne appartient à \mathcal{C} si et seulement si il existe au moins deux bases B_i et B_j de \mathcal{D} telles que le signe de ces deux bases dans M est $-$ (i.e. $\chi_M(B_i) = \chi_M(B_j) = -$). De manière équivalente un matroïde orienté M associé à un demi-crâne appartient à $\overline{\mathcal{C}}$ si et seulement si il existe au moins deux bases B_i et B_j de \mathcal{D} telles que le signe de ces deux bases dans M est $+$ (i.e. $\chi_M(B_i) = \chi_M(B_j) = +$). En reprenant les notations de la Section ??, on a $\mathcal{C} = \mathcal{C}_{\mathcal{D},\{-,-,-\},2}$ et $\overline{\mathcal{C}} = \mathcal{C}'_{\mathcal{D},\{-,-,-\},2}$.

Il n'est pas surprenant qu'il n'existe pas de bases discriminantes entre le cluster constitué des demi-crânes des Unaffected (L-Unaffected et R-Unaffected) et les trois autres clusters à un taux de

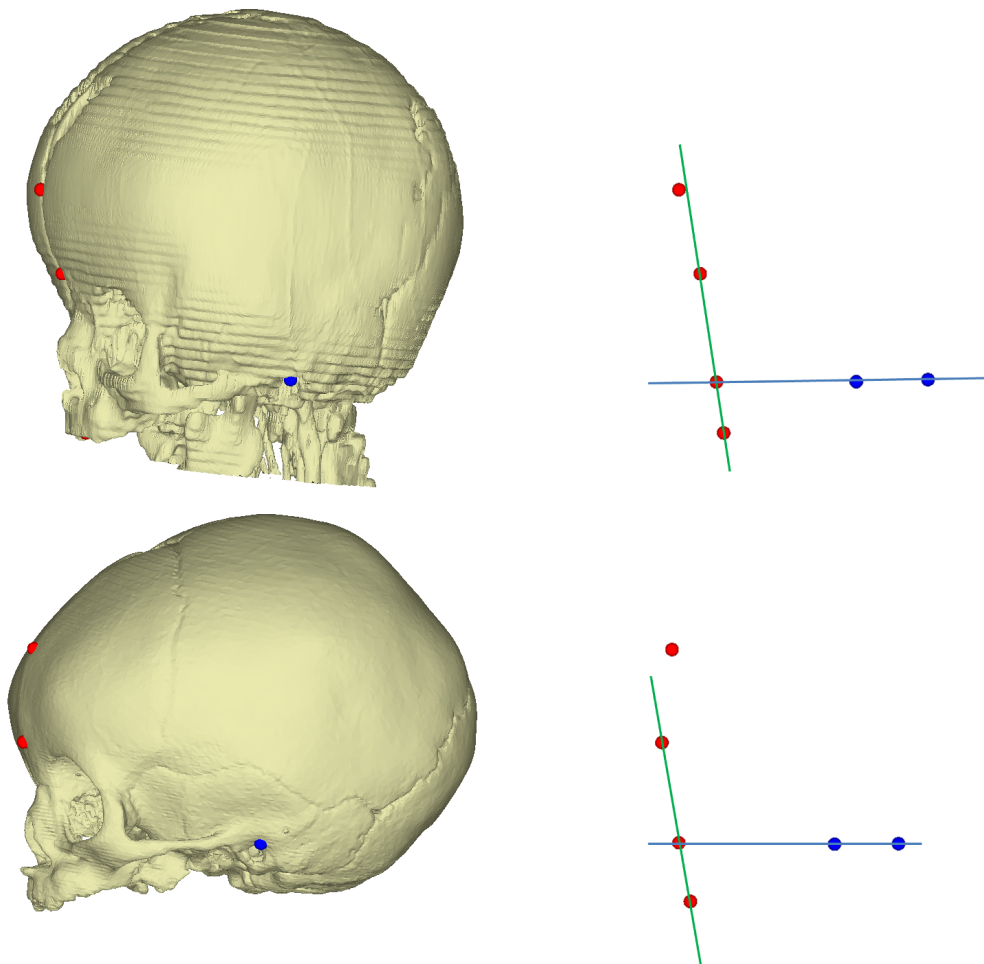


FIGURE 11.15 – Exemple d'un crâne BCS (en haut) et d'un crâne Unaffected (en bas). Les images de droite représentent la position du semi-landmarks situé sur l'avant de la voûte crânienne par rapport au plan (en vert) formé par les trois semi-landmarks situés sur la face du crâne. Le plan en bleu nous sert ici de plan de référence et relie les deux oreilles internes (points bleus) au semi-landmark de l'orbite droite.

100% puisque lors de notre étude sur les crânes en entier la classe des Unaffected était la classe ayant le moins de bases possédant un taux de 100%.

Discussion

Grâce aux résultats de l'Analyse en Composantes Principales (Figure 11.12) et aux distances de Procrustes (Table 11.5), Heuzé et al. ont conclu que les crânes des BCS et des individus non-affectés ont un faible degré d'asymétrie. D'après les auteurs, ces résultats valideraient l'hypothèse H1 selon laquelle lorsque la fusion de la suture à lieu à la fois sur la partie droite et sur la partie gauche du crâne alors la déformation affecte les deux côtés de façon similaire. Ces résultats indiquent également que les deux classes de demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture (R-LUCS et L-RUCS) sont proches et qu'il en est de même pour les deux classes de demi-crânes des LUCS et RUCS dont la suture est fusionnée (R-RUCS et L-LUCS). Les auteurs en ont conclu que les classes LUCS et RUCS seraient des images miroirs l'une de l'autre (c'est-à-dire que l'hypothèse H2 serait validée et qu'il n'y aurait qu'un seul processus de déformation).

Tout comme dans le tableau calculé par Heuzé et al. (11.5), on remarque avec notre méthode que les demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture ({R-LUCS, L-RUCS}) sont légèrement plus proche des demi-crânes Unaffected que les demi-crânes des LUCS et RUCS dont la suture est fusionnée ({R-RUCS, L-LUCS}). Ainsi, la moyenne des distances entre un chirotope d'un demi-crâne R-LUCS et un chirotope d'un demi-crâne Unaffected est de 26,3% si ce demi-crâne est un demi-crâne gauche et est de 25,1% si ce demi-crâne est un demi-crâne droit. Par contre la moyenne des distances entre un chirotope d'un demi-crâne L-LUCS et un chirotope d'un demi-crâne Unaffected varie de 27% (s'il s'agit d'un demi-crâne gauche), à 27,5% (s'il s'agit d'un demi-crâne droit). Il semblerait donc que les demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture seraient moins déformés que les demi-crânes des LUCS et RUCS dont la suture est fusionnée.

Dans la partition obtenue à partir de l'algorithme des k-means, les demi-crânes gauches et droits des BCS (R-BCS et L-BCS) forment un cluster ce qui validerait l'hypothèse H1 selon laquelle lorsque la fusion de la suture à lieu à la fois sur la partie droite et sur la partie gauche du crâne alors la déformation affecte les deux côtés de façon similaire.

De même, deux autres clusters sont formés respectivement par les demi-crânes des LUCS et RUCS ne présentant pas de fusion de la suture ({R-LUCS, L-RUCS}) et par les demi-crânes des LUCS et RUCS dont la suture est fusionnée ({R-RUCS, L-LUCS}). Ainsi les classes LUCS et RUCS semble bien être des images miroirs l'une de l'autre ce qui validerait l'hypothèse H2. Par contre comme la distance entre le cluster formé par les demi-crânes des BCS et le cluster formé par les demi-crânes des LUCS et RUCS dont la suture est fusionnée ({R-RUCS, L-LUCS}) n'est pas nettement inférieure aux autres distances, rien ne nous permet de conclure que le processus de déformation est le même sur les côtés des crânes présentant une fusion de la suture. Notre étude ne permet donc pas de vérifier l'hypothèse H3.

Nous avons trouvé des bases discriminantes à un taux de 100% pour les trois premiers clusters et une caractérisation des demi-crânes Unaffected n'utilisant que trois bases. Ceci nous indique que ces clusters ont une réelle signification au niveau de la position relative des landmarks et semi-landmarks. Ces résultats sont donc en adéquation avec les hypothèses H1 et H2 selon lesquelles

lorsque la fusion de la suture à lieu à la fois sur la partie droite et sur la partie gauche du crâne alors la déformation affecte les deux côtés de façon similaire (H1) et que les classes LUCS et RUCS sont des images miroirs l'une de l'autre (H2).



Conclusion et perspectives de recherche

Dans cette thèse, nous sommes partis du constat que les méthodes de morphométrie actuelle fondées sur les landmarks ne permettaient pas toujours de mettre en valeur des différences “structurelles”, c’est-à-dire des variations de la structure des formes indépendamment de leurs intensités. Or, dans de nombreux cas, ce sont ces différences structurelles qui caractérisent la structure anatomique.

Pour cela, dans la Partie I, nous avons présenté une partie de la théorie des matroïdes orientés qui est étudiée en mathématiques mais peu utilisée dans des applications pratiques. Dans cette théorie, on code un ensemble de n points en dimension d en associant un signe 0, + ou - à chaque $(d+1)$ -uplet de points. On obtient alors une liste de signes appelé chirotope. Les différences structurelles vont alors se caractériser par des changements de signes. Ainsi notre méthode consiste à analyser les chirotopes afin de déterminer les changements de signes significatifs de $(d+1)$ -uplets de landmarks.

Dans la Partie II, nous nous sommes intéressés à des $(d+1)$ -uplets de points en dimension d pour lesquels, pour chaque coordonnée de l’espace, un ordre est fixé entre les différents sommets.

Tout d’abord, dans le Chapitre 3, nous avons voulu déterminer à partir d’un ensemble d’ordres sur les coordonnées des points, si le signe associé à ce $(d+1)$ -uplet de points est constant quelles que soient les coordonnées des points. L’intérêt de ceci est de trouver des $(d+1)$ -uplets qui seront toujours fixes à partir d’hypothèses, afin de les éliminer de l’étude de structures anatomiques. Nous avons établi les Théorèmes 3.8 et 3.10 qui nous donnent respectivement en dimension deux et trois une caractérisation des ensembles d’ordres tels que le signe associé aux $(d+1)$ -uplets de points satisfaisant ces ensembles d’ordres est constant. Ces résultats ont été présentés à la conférence internationale CCCG’11 [21] et soumis à la revue *Linear Algebra and its Applications* en mai 2013.

Ces résultats soulèvent une première question : *existe-t-il également une caractérisation de ces ensembles d’ordres en dimensions supérieures ?*

Dans le Chapitre 4, nous nous sommes intéressés à un problème dérivé de la démonstration du Chapitre 3, qui est de trouver le nombre d’ensembles d’ordres totaux équivalents à transformations près, en fonction de la dimension. Nous avons expliqué en particulier comment fonctionne

le programme qui nous a permis d'obtenir ce nombre pour des petites dimensions. Ce programme génère des ensembles d'ordres totaux puis supprime ceux qui sont égaux à transformations près à un ensemble d'ordres totaux déjà généré.

On peut alors se demander : *existe-t-il une fonction mathématique permettant de calculer directement le nombre d'ensembles d'ordres totaux équivalents à transformations près ?*

Dans le Chapitre 5, nous avons étudié certaines méthodes de classification déjà existantes afin de déterminer si elles peuvent être appliquées à un ensemble de chirotopes et si elles peuvent nous fournir une caractérisation des classes. Pour utiliser ces méthodes, il était nécessaire de transformer les chirotopes en données binaires ou numériques. Dans la Section 5.1.2 nous avons alors représenté un chirotope par un vecteur de $\{-1, 1\}^l$ où l est le nombre de $(d+1)$ -uplet de points si les points sont en position générale et par un vecteur de $\{0, 1\}^{3l}$ sinon.

Cependant, chaque vecteur de $\{-1, 1\}^l$ ou de $\{0, 1\}^{3l}$ ne correspond pas nécessairement à un chirotope associé à un ensemble de points. Une question naturelle est alors la suivante : *comment trouver un vecteur de $\{-1, 1\}^l$ (resp. de $\{0, 1\}^{3l}$) correspondant à un chirotope, le plus proche possible d'un vecteur donné de $\{-1, 1\}^l$ (resp. de $\{0, 1\}^{3l}$) ?* Cette question semble difficile puisqu'elle est liée au problème de déterminer si un matroïde orienté est réalisable ou non. Or ce problème est équivalent à un problème NP-difficile [72]. Nos applications étant en 2D et 3D, on pourrait contourner le problème de réalisabilité en générant toutes les configurations de points 2D et 3D. Pour l'instant de tels énumérations n'ont été possible que pour un nombre de points inférieur ou égal à 9 en 2D et à 8 en 3D [19].

Pour essayer de simplifier la question précédente, on peut alors essayer de partir d'un chirotope proche d'un vecteur donné de $\{-1, 1\}^l$ (resp. de $\{0, 1\}^{3l}$) et on cherche à s'en rapprocher encore plus, en faisant traverser certains plans à certains points. Cela pose de nouvelles questions : *Comment peut-on déterminer les plans et les points qui doivent les traverser ? Si l'on doit faire traverser plusieurs plans, peut-on les faire traverser simultanément ou l'ordre a-t-il de l'importance ?*

Lors de notre analyse des méthodes de classification, nous avons décidé d'utiliser l'algorithme des k -means (voir Section 5.2.2). L'inconvénient de cet algorithme est de classer les chirotopes à partir de points moyens qui n'appartiennent ni à $\{-1, 1\}^l$ dans le cas où les points sont en position générale, ni à $\{0, 1\}^{3l}$ dans le cas contraire. Pour pallier ce problème, on pourrait remplacer à chaque étape de l'algorithme, chaque point moyen par un vecteur correspondant à un chirotope le plus proche possible de ce point moyen. Cette méthode nécessite de savoir répondre à la question précédente mais pose également d'autres questions : *si l'on remplace à chaque étape de l'algorithme des k -means chaque point moyen par un vecteur correspondant à un chirotope le plus proche possible de ce point moyen, l'algorithme obtenu converge-t-il ? Si oui quels sont les propriétés du résultat obtenu et est-il proche de celui de l'algorithme des k -means ?* Cette méthode semble proche de l'algorithme des k -medoids qui utilise le même principe que dans les k -means pour la recherche de meilleurs médoïdes [54], mais contrairement à cet algorithme nous ne connaissons pas l'ensemble de tous les chirotopes réalisables.

Comme il semble difficile de trouver un vecteur correspondant à un chirotope le plus proche possible d'un vecteur donné, on pourrait se contenter d'arrondir chaque coordonnée des points moyens à chaque étape de l'algorithme des k -means. Il se pose alors les mêmes questions que précédemment, à savoir : *si l'on arrondit chaque coordonnée des points moyens à chaque étape*

de l'algorithme des k -means, l'algorithme obtenu converge-t-il? Si oui quels sont les propriétés du résultat obtenu et est-il proche de celui de l'algorithme des k -means?

Dans le Chapitre 6, nous proposons deux méthodes de caractérisation de classes de chirotopes : la caractérisation forte et la caractérisation faible. Ces méthodes consistent à caractériser chaque classe sous forme d'une "boule" définie sur un certain sous-ensemble \mathcal{D} de $(d+1)$ -uplets de points. Pour cela, pour chaque classe \mathcal{C} nous trions tous les $(d+1)$ -uplets selon leur taux de discriminabilité entre \mathcal{C} et son complémentaire $\overline{\mathcal{C}}$. Plus un $(d+1)$ -uplet de points a un fort taux de discriminabilité entre \mathcal{C} et $\overline{\mathcal{C}}$ et plus ce $(d+1)$ -uplet permet de séparer \mathcal{C} de $\overline{\mathcal{C}}$. Ainsi pour chaque classe, nous cherchons le sous-ensemble \mathcal{D} parmi les $(d+1)$ -uplets de points ayant un fort taux de discriminabilité. Par contre, le calcul des distances n'est pas pondéré par le taux de discriminabilité. On peut alors se demander : *comment modifier le calcul des distances afin qu'elles prennent en compte le taux de discriminabilité?*

Dans le Chapitre 7, nous avons présenté l'algorithme permettant de calculer le chirotope d'une configuration de points en dimension quelconque. Nous avons vu que le temps de calcul pour obtenir l'ensemble des chirotopes pouvait être important (de l'ordre de 4h30 pour les 60 échantillons composés chacun de 133 landmarks). Il se pose alors une nouvelle question : *comment réduire le temps de calcul des chirotopes?* On pourrait par exemple gagner facilement du temps en calculant en parallèle les chirotopes de plusieurs échantillons puisque les procédures sont indépendantes.

Nous avons également expliqué la méthode employée pour garantir l'exactitude des calculs. Par contre, notre algorithme ne prend pas en compte l'incertitude sur les données qui peut provenir entre autre du relevé approximatif des landmarks [75]. Par exemple dans [71], les auteurs ont mesuré que la position d'un même landmark anatomique relevé par plusieurs spécialistes sur un même crâne pouvait varier jusqu'à 1,81 mm. L'incertitude sur les données provient également de l'utilisation de flottants pour stocker les coordonnées des landmarks. Or une variation même très faible dans les coordonnées d'un landmarks peut changer le signe de $(d+1)$ -uplets de points. Cela soulève alors la question suivante : *comment gérer l'incertitude sur les données?* Pour répondre à cette question, il pourrait être utile d'utiliser l'arithmétique par intervalles [85] qui consiste à remplacer chaque nombre par un intervalle. Ce principe a notamment été utilisé dans [60]. On pourrait alors déterminer si le signe d'un $(d+1)$ -uplet de points est constant quelles que soient les valeurs prises dans ces intervalles. Par contre cela soulève une autre question : *si l'imprécision sur les données peut modifier le signe d'un $(d+1)$ -uplet de points, doit-on mettre ce signe à 0 ou doit-on considérer ces signes différemment des signes pour lesquels il n'y a pas d'incertitude?*

Dans la Partie V, nous avons présenté les résultats obtenus en appliquant notre méthode à des jeux de données anatomiques. Dans le Chapitre 9, nous avons étudié la forme du crânes de 182 humains et de 108 grands singes (58 bonobos et 50 chimpanzés). Nous avons utilisé l'algorithme des k -means pour obtenir une classification des chirotopes en deux clusters. Seulement deux humains et deux grands singes sont mal classés par cet algorithme ce qui fait que 99% des chirotopes sont bien classés. Ainsi une question se pose alors : *pourquoi ces quatre chirotopes sont mal classés par l'algorithme des k -means lorsque l'on cherche à classer les chirotopes en deux clusters?* En répondant à cette question nous pourrions déterminer si ces données sont atypiques ou s'il nous faut développer de nouvelles méthodes de classification permettant de classer parfaitement ces données.

Nous avons également cherché à caractériser les humains et les grands singes. Nous n'avons pas pu caractériser les deux espèces à cause de certains chirotopes. Par exemple, la caractérisation

de ces espèces à l'aide d'une seule base n'est pas possible à cause d'un humain et de trois grands singes. Il se pose alors une question analogue à la question précédente : *pourquoi cet humain et ces trois grands singes ne nous permettent pas d'obtenir une caractérisation à l'aide d'une seule base ?*

Dans le Chapitre 10, nous avons étudié les classes dentaires (classe I, classe II et classe III) utilisées en orthodontie. Nous avons calculé les moyennes des distances entre les chirotopes en fonction de ces classes. D'après ces moyennes, il semble que les chirotopes des classes I et III sont semblables et que les chirotopes de ces deux classes se différencient de ceux de la classe II. Nous avons alors utilisé l'algorithme des k -means pour obtenir une classification des chirotopes en deux clusters afin de voir si cette classification sépare les chirotopes de la classe II de ceux des autres classes. Dans la classification obtenue, un cluster correspond bien aux chirotopes de la classe II et l'autre cluster correspond aux chirotopes des classes I et III sauf pour 9% des données. Ainsi une question se pose alors : *pourquoi ces données sont mal classées par l'algorithme des k -means lorsque l'on cherche à classer les chirotopes en deux clusters et pourrait-on développer de nouvelles méthodes de classification afin d'augmenter le taux de réussite de la classification ?* Nous n'avons pas pu caractériser l'ensemble des chirotopes appartenant à la classe II et l'ensemble des chirotopes appartenant aux deux autres classes. En étudiant les modèles numériques tridimensionnels, nous avons remarqué que de nombreux individus présentent de lourdes déformations crâniennes. La présence de ces déformations soulève alors une autre question : *ces déformations expliquent-elles l'absence de la convexité faciale pour certains individus de la classe II ou est-ce que chez certains individus de la classe II les dents sont décalées sans qu'il y ait un recul du maxillaire inférieur ?*

Enfin dans le Chapitre 11, nous avons étudié les déformations cranio-faciales que présentent les enfants atteints de craniosynostose coronale. Dans [31], les auteurs ont étudié cette base de données en utilisant des méthodes classiques de morphométrie.

Nous avons dans un premier temps étudié ces données en utilisant l'ensemble des landmarks et semi-landmarks relevés par les experts. Nous avons utilisé l'algorithme des k -means pour obtenir une classification des chirotopes en quatre clusters. Les clusters fournis par cet algorithme correspondent exactement aux quatre classes définies par les experts. Nous avons ensuite cherché à caractériser ces classes. Nous avons trouvé plusieurs bases totalement discriminantes permettant de caractériser chaque classe (de 7 bases pour la classe Unaffected à 15 667 pour la classe LUCS). Ainsi l'étude de la position relative de ces landmarks et semi-landmarks permet de déterminer à quelle classe appartient un individu.

Nous avons ensuite étudié ces données en utilisant uniquement les landmarks anatomiques. Les quatre clusters fournis par l'algorithme des k -means correspondent exactement aux quatre classes définies par les experts à une exception près : un individu de la classe BCS est classé avec ceux appartenant à la classe LUCS. Cet individu est étonnant car si l'on utilise l'algorithme des k -medoids pour répartir les individus en quatre clusters alors cet individu est classé cette fois-ci avec les individus de la classe RUCS. Il se pose alors une question : *pourquoi cet individu est mal classé par ces méthodes et pourquoi son classement varie selon la méthode utilisée ?* Nous avons ensuite cherché à caractériser ces classes. Nous avons obtenu une caractérisation forte utilisant deux bases pour la classe BCS et une caractérisation forte utilisant cinq bases pour la classe Unaffected. Nous avons également trouvé 22 bases totalement discriminantes pour les LUCS et 4 bases totalement discriminantes pour les RUCS. De plus, pour chaque base B parmi les 4 bases totalement discriminantes pour les RUCS, si l'on remplace les landmarks de B par les landmarks symétriques par

rapport au plan médian sagittal alors la base obtenue est totalement discriminantes pour les LUCS. Cela suggère que les classes RUCS et LUCS sont des images miroirs l'une de l'autre.

Dans [31], Heuzé et al. ont étudié l'asymétrie des déformations de crânes. Nous avons réalisé une étude de l'asymétrie en nous basant sur les chirotopes de chaque demi-crâne. Nous avons alors obtenu une partition des chirotopes par l'algorithme des k -means, puis nous avons déterminé une caractérisation de chaque cluster. Ces résultats nous amènent aux mêmes conclusions que dans [31] : la déformation des crânes de la classe BCS conserve la symétrie bilatérale et les classes RUCS et LUCS sont des images miroirs l'une de l'autre.

Lors de cette thèse, nous avons mis en place une méthodologie basée sur les positions relatives de landmarks permettant de comparer et de classer des échantillons, ainsi que de caractériser des classes d'échantillons. Nous avons retrouvé par notre méthode des résultats connus, ce qui indique que notre méthode mériterait d'être utilisée à l'instar des méthodes classiques de morphométrie. De plus, nous avons caractérisé les classes d'échantillons par des propriétés anatomiques, confirmant l'intérêt de notre méthode. Ainsi, on peut espérer découvrir de nouvelles propriétés anatomiques en appliquant notre méthode à d'autres jeux de données.



Bibliographie

- [1] D.C. Adams, E.J. Rohlfs, and D.E. Slice. Geometric morphometrics : ten years of progress following the “revolution”. *Italian Journal of Zoology*, 71(1) :5–16, 2004. Cité page 10.
- [2] D. Aloise, A. Deshpande, P. Hansen, and P. Popat. Np-hardness of euclidean sum-of-squares clustering. *Machine Learning*, 75(2) :245–248, May 2009. Cité page 94.
- [3] D. Arthur and S. Vassilvitskii. k-means++ : the advantages of careful seeding. In *SODA '07 : Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics. Cité page 96.
- [4] A. Bachem and W. Kern. *Linear Programming Duality : An Introduction to Oriented Matroids*. Universitext (1979). Springer-Verlag, 1992. Cité page 21.
- [5] A. Björner, M. Las Vergnas, B. Sturmfels, N. White, and G.M. Ziegler. *Oriented Matroids*. Cambridge U. Press, Encyclopedia of Mathematics, Cambridge UK 1993, 1993. Cité pages 21, 33 et 40.
- [6] F. Bookstein. Principal warps : Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(6) :567–585, 1989. Cité page 14.
- [7] F. Bookstein. *Morphometric tools for landmark data : geometry and biology*. Cambridge University Press, June 1997. Cité page 7.
- [8] J. Braga and J. Treil. Estimation of pediatric skeletal age using geometric morphometrics and three-dimensional cranial size changes. *International Journal of Legal Medicine*, 121 :439–443, 2007. 10.1007/s00414-007-0170-x. Cité pages 42, 43, 151, 205 et 206.
- [9] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Chapman and Hall/CRC, 1 edition, January 1984. Cité page 105.
- [10] R. Brualdi and B. Shader. *Matrices of Sign-Solvable Linear Systems*. Cambridge Tracts in Mathematics. Cambridge University Press, 2009. Cité page 40.

- [11] A. Cheetham and J. Hazel. Binary (presence-absence) similarity coefficients. *Journal of Paleontology*, 43(5) :1130–1136, 1969. Cité page 83.
- [12] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(8) :790–799, August 1995. Cité pages 101 et 102.
- [13] T. Cole III and J. Richtsmeier. A simple method for visualization of influential landmarks when using euclidean distance matrix analysis. *Am J Phys Anthropol*, 107(3) :273–83, 1998. Cité page 18.
- [14] A. Dress, A. Dreiding, and H. Haegi. Classification of mobile molecules by category theory. In *Symmetries and Properties of Non-Rigid Molecules : A Comprehensive Survey*, volume 23, pages 39–58, 1983. Cité page 21.
- [15] R. Dubes and A. Jain. Clustering methodologies in exploratory data analysis. *Advances in Computers*, 19 :113–228, 1980. Cité page 95.
- [16] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231, 1996. Cité pages 101 et 102.
- [17] S. Felsner. On the number of arrangements of pseudolines. In *Proceedings of the twelfth annual symposium on Computational geometry*, SCG '96, pages 30–37, New York, NY, USA, 1996. ACM. Cité page 21.
- [18] J. Folkman and J. Lawrence. Oriented matroids. *J. Comb. Theory, Ser. B*, 25(2) :199–236, 1978. Cité page 21.
- [19] K. Fukuda, H. Miyata, and S. Moriyama. Complete enumeration of small realizable oriented matroids. *Discrete & Computational Geometry*, 49(2) :359–381, 2013. Cité page 190.
- [20] E. Gioan and J. Ramirez-Alfonsin. Eléments de théorie des matroïdes et matroïdes orientés. In Philippe Langlois, editor, *Informatique Mathématique - Une photographie en 2013*, pages 47–95. Presses Universitaires de Perpignan, April 2013. Cité pages 21 et 32.
- [21] E. Gioan, K. Sol, and G. Subsol. Orientations of simplices determined by orderings on the coordinates of their vertices. In *CCCG*, pages 187–192, 2011. Cité pages 4, 39 et 189.
- [22] E. Gioan, K. Sol, and G. Subsol. Sequence a201973 in the on-line encyclopedia of integer sequences. published electronically at, 2011. Cité pages 52, 63 et 76.
- [23] E. Gioan, K. Sol, and G. Subsol. A combinatorial method for 3d landmark-based morphometry : Application to the study of coronal craniosynostosis. In Nicholas Ayache, Hervé Delingette, Polina Golland, and Kensaku Mori, editors, *MICCAI (3)*, volume 7512 of *Lecture Notes in Computer Science*, pages 533–541. Springer, 2012. Cité pages 4, 40, 111, 167 et 170.
- [24] E. Gioan, K. Sol, G. Subsol, Y. Heuzé, J. Richtsmeier, J. Braga, and F. Thackeray. A new 3d morphometric method based on a combinatorial encoding of 3d point configurations : application to skull anatomy for clinical research and physical anthropology. In *Poster : 80th Annual Meeting of the American Association of Physical Anthropologists, Minneapolis (U.S.A.), April 2011. Abstract : American Journal of Physical Anthropology, p. 280, Vol. 144 Issue S52*, 2011. Cité pages 4, 40, 43 et 167.

- [25] Emeric Gioan, Kevin Sol, Gérard Subsol, Jose Braga, and Jacques Treil. Une nouvelle méthode de morphométrie 3d par codage combinatoire de configurations de points 3d : application à l'anatomie du crâne. In *18ièmes Journées de la Société d'Anthropologie de Paris*, France, January 2011. Cité page 4.
- [26] L.W. Graber, R.L. Vanarsdall, and K.W.L. Vig. *Orthodontics : Current Principles and Techniques*. Elsevier Health Sciences, 2011. Cité page 159.
- [27] U. Grenander and M. Miller. Computational anatomy : an emerging discipline. *Q. Appl. Math.*, LVI(4) :617–694, 1998. Cité page 17.
- [28] G. Hamerly and C. Elkan. Alternatives to the k-means algorithm that find better clusterings. In *Proceedings of the eleventh international conference on Information and knowledge management, CIKM '02*, pages 600–607, New York, NY, USA, 2002. ACM. Cité page 96.
- [29] G. Hamerly and C. Elkan. Learning the k in k-means. In *Advances in neural information processing systems 16 : proceedings of the 2003 conference*, volume 16, page 281. The MIT Press, 2004. Cité page 96.
- [30] J. Han, M. Kamber, and A. Tung. Spatial clustering methods in data mining : A survey. In Harvey J. Miller and Jiawei Han, editors, *Geographic Data Mining and Knowledge Discovery, Research Monographs in GIS*. Taylor and Francis, 2001. Cité page 98.
- [31] Y. Heuzé, N. Martínez-Abadías, J. Stella, C. Senders, S. Boyadjiev, L. Lo, and J. Richtsmeier. Unilateral and bilateral expression of a quantitative trait : asymmetry and symmetry in coronal craniosynostosis. *Journal of Experimental Zoology Part B : Molecular and Developmental Evolution*, 318(2) :109–122, 2012. Cité pages 166, 176, 177, 179, 192, 193 et 207.
- [32] M. Inaba, N. Katoh, and H. Imai. Applications of weighted voronoi diagrams and randomization to variance-based k-clustering : (extended abstract). In *Proceedings of the tenth annual symposium on Computational geometry, SCG '94*, pages 332–339, New York, NY, USA, 1994. ACM. Cité page 94.
- [33] P. Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37 :547–579, 1901. Cité page 83.
- [34] A. Jain and R. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988. Cité page 101.
- [35] E. Kaltofen and G. Villard. Computing the sign or the value of the determinant of an integer matrix, a complexity survey. *Journal of Computational and Applied Mathematics*, 162(1) :133–146, 2004. Cité page 136.
- [36] L. Kaufman and P.J. Rousseeuw. Clustering by means of medoids. In Y. Dodge, editor, *Statistical Data Analysis Based on the L1-Norm and Related Methods*, pages 405–416. North-Holland, Amsterdam, 1987. Cité page 97.
- [37] L. Kaufman and P.J. Rousseeuw. *Finding groups in data : an introduction to cluster analysis*. Wiley series in probability and mathematical statistics : Applied probability and statistics. John Wiley and Sons, New York, 1990. Cité page 99.

- [38] H. Kriegel, P. Kröger, J. Sander, and A. Zimek. Density-based clustering. *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery*, 1(3) :231–240, 2011. Cité page 101.
- [39] D.C. Lay and M. Citta-Vanthsche. *Algèbre linéaire : Théorie, exercices & applications*. De Boeck, 2004. Cité page 136.
- [40] S. Lele and J. T. Richtsmeier. Euclidean distance matrix analysis : a coordinate-free approach for comparing biological shapes using landmark data. *Am J Phys Anthropol*, 86(3) :415–427, November 1991. Cité page 15.
- [41] M. Levandowsky and D. Winter. Distance between sets. *Nature*, 234(5323) :34–35, 1971. Cité page 83.
- [42] T. Li. A general model for clustering binary data. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 188–197, New York, NY, USA, 2005. ACM. Cité page 102.
- [43] A. Likas, N. Vlassis, and J. Verbeek. The global k-means clustering algorithm. *Pattern Recognition*, 36(2) :451–461, February 2003. Cité page 96.
- [44] Richa Loochach and Kanwal Garg. Effect of distance functions on simple k-means clustering algorithm. *International Journal of Computer Applications*, 49(6) :7–9, July 2012. Published by Foundation of Computer Science, New York, USA. Cité page 96.
- [45] L. Marcus. *Proceedings of the Michigan Morphometric Workshop. Special Publication No. 2.*, chapter Chapter 4. Traditional morphometrics, pages 77–122. Ann Arbor MI, The University of Michigan Museum of Zoology, 1990. Cité page 11.
- [46] D. Michie, D. Spiegelhalter, C. Taylor, and J. Campbell. *Machine learning, neural and statistical classification*. Ellis Horwood, Upper Saddle River, NJ, USA, 1994. Cité page 109.
- [47] E.M. Mirkes. K-means and k-medoids applet, 2011. Cité pages 99 et 100.
- [48] P. Mitteroecker and P. Gunz. Advances in geometric morphometrics. *Evolutionary Biology*, 36 :235–247, 2009. Cité pages 10 et 11.
- [49] J.P. Nakache and J. Confais. *Approche pragmatique de la classification : Arbres hiérarchiques, Partitionnements*. Technip, 2004. Cité pages 83, 87, 93 et 102.
- [50] R. Ng and J. Han. Efficient and effective clustering methods for spatial data mining. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 144–155, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc. Cité page 99.
- [51] Jianwei Niu, Zhizhong Li, and Gavriel Salvendy. Mathematical methods for shape analysis and form comparison in 3d anthropometry : a literature review. In *Proceedings of the 1st international conference on Digital human modeling*, ICDHM'07, pages 161–170, Berlin, Heidelberg, 2007. Springer-Verlag. Cité page 10.
- [52] P. O'Higgins. The study of morphological variation in the hominid fossil record : biology, landmarks and geometry. *Journal of Anatomy*, 197 :103–120, 2000. Cité pages 7 et 13.

- [53] R. Pan, F. Wei, and M. Li. Craniofacial variation of the chinese macaques explored with morphologica. *Journal of Morphology*, 256(3) :342–348, 2003. Cité page 13.
- [54] Hae-Sang Park and Chi-Hyuck Jun. A simple and fast algorithm for k-medoids clustering. *Expert Syst. Appl.*, 36(2) :3336–3341, March 2009. Cité pages 99, 100, 190 et 206.
- [55] S. Peck. A biographical portrait of edward hartley angle, the first specialist in orthodontics. *Angle Orthod*, 79(6) :1021–1036, 2009. Cité page 159.
- [56] D. Pelleg and A. Moore. X-means : Extending k-means with efficient estimation of the number of clusters. In *Proceedings of the Seventeenth International Conference on Machine Learning, ICML '00*, pages 727–734, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. Cité page 96.
- [57] X. Pennec and P. Fillard. Statistical computing on non-linear spaces for computational anatomy. In Nikos Paragios, Jim Duncan, and Nicholas Ayache, editors, *Biomedical Image Analysis : Methodologies And Applications*. Springer, 2013. To Appear. Cité page 17.
- [58] J. Pfeifle and J. Rambau. Computing triangulations using oriented matroids. Technical Report 02-02, ZIB, Takustr.7, 14195 Berlin, 2002. Cité pages 21 et 133.
- [59] M. Pichat and J. Vignes. *Ingénierie du contrôle de la précision des calculs sur ordinateur*. Collection Informatique. Éditions Technip, 1993. Cité page 35.
- [60] S. Pion. *De la géométrie algorithmique au calcul géométrique*. PhD thesis, Univ. de Nice, Sophia-Antipolis, Rocquencourt, 1999. Th. : informatique. Cité page 191.
- [61] M. Pocchiola and G. Vegter. Pseudo-triangulations : theory and applications. In *Proceedings of the twelfth annual symposium on Computational geometry, SCG '96*, pages 291–300, New York, NY, USA, 1996. ACM. Cité page 21.
- [62] D. Proctor, D. Broadfield, and K. Proctor. Quantitative three-dimensional shape analysis of the proximal hallucial metatarsal articular surface in homo, pan, gorilla, and hylobates. *American Journal of Physical Anthropology*, 135(2) :216–224, 2008. Cité page 13.
- [63] A. Pullinger, W. Solberg, L. Hollender, and A. Petersson. Relationship of mandibular condylar position to dental occlusion factors in an asymptomatic population. *American Journal of Orthodontics and Dentofacial Orthopedics*, 91(3) :200 – 206, 1987. Cité page 159.
- [64] R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1) :81–106, 1986. Cité page 105.
- [65] R. Quinlan. *C4.5 : programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. Cité page 105.
- [66] C. Radhakrishna Rao. The utilization of multiple measurements in problems of biological classification. *Journal of the Royal Statistical Society - Series B*, 10(2) :159–203, 1948. Cité page 83.
- [67] J. Richter-Gebert and G.M. Ziegler. Oriented matroids. In Jacob E. Goodman and Joseph O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 111–132, Boca Raton-New York, 1997. CRC Press. Cité page 21.

- [68] J. Richtsmeier, V. Deleon, and S. Lele. The promise of geometric morphometrics. *American Journal of Physical Anthropology*, 119(S35) :63–91, 2002. Cité pages 10, 17 et 18.
- [69] J.T. Richtsmeier and S. Lele. A coordinate-free approach to the analysis of growth patterns : models and theoretical considerations. *Biol Rev Camb Philos Soc*, 68(3) :381–411, 1993. Cité page 16.
- [70] J. Sander, M. Ester, H. Kriegel, and X. Xu. Density-based clustering in spatial databases : The algorithm gbscan and its applications. *Data Mining and Knowledge Discovery*, 2 :169–194, 1998. 10.1023/A :1009745219419. Cité page 101.
- [71] W. Schlicher, I. Nielsen, J.C. Huang, K. Maki, D.C. Hatcher, and A.J. Miller. Consistency and precision of landmark identification in three-dimensional cone beam computed tomography scans. *Eur J Orthod*, 34(3) :263–75, 2012. Cité page 191.
- [72] P. Shor. Stretchability of pseudolines is np-hard. Applied geometry and discrete mathematics, Festschr. 65th Birthday Victor Klee, DIMACS, Ser. Discret. Math. Theor. Comput. Sci. 4, 531-554 (1991)., 1991. Cité page 190.
- [73] R.R. Sokal and P.H.A. Sneath. *Principles of numerical taxonomy*. Series of books in biology. W. H. Freeman, 1963. Cité page 83.
- [74] E. Staffetti, A. Grau, F. Serratos, and A. Sanfeliu. Object and image indexing based on region connection calculus and oriented matroid theory. *Discrete Appl. Math.*, 147(2-3) :345–361, 2005. Cité page 21.
- [75] G. Subsol. Le problème de la définition des repères 3d pour l'analyse morphométrique en anthropologie physique. *Biométrie humaine et anthropologie*, Tome 30, 1-2 :37–45, 2012. Cité pages 7 et 191.
- [76] J. Sun and H. Li. Data mining method for listed companies' financial distress prediction. *Knowledge-Based Systems*, 21(1) :1 – 5, 2008. Cité pages 105 et 106.
- [77] A.J. Surkan and J.C. Singleton. Neural networks for bond rating improved by multiple hidden layers. In *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, pages 157–162 vol.2, 1990. Cité page 109.
- [78] P. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison-Wesley, 2005. Cité page 87.
- [79] J. Treil, J. Braga, and A. Aït Ameer. Modélisation 3d du viscéro-crâne. applications en orthodontie et chirurgie orthognatique. *Journal de Radiologie*, 90(5, Part 2) :634 – 641, 2009. Cité page 159.
- [80] M. Van der Laan, K. Pollard, and J. Bryan. A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8) :575–584, August 2003. Cité page 99.
- [81] S. Visa, A. Ralescu, and M. Ionescu. Investigating learning methods for binary data. In *Fuzzy Information Processing Society, 2007. NAFIPS '07. Annual Meeting of the North American*, pages 441–445, 2007. Cité page 109.

- [82] N. von Cramon-Taubadel, B. Frazier, and M. Lahr. The problem of assessing landmark error in geometric morphometrics : Theory, methods, and modifications. *Am J Phys Anthropol*, 134 :24–35, 2007. Cité page 17.
- [83] S. J. Wan, S. K. M. Wong, and P. Prusinkiewicz. An algorithm for multidimensional data clustering. *ACM Trans. Math. Softw.*, 14(2) :153–162, June 1988. Cité page 94.
- [84] Jr. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301) :236–244, March 1963. Cité page 91.
- [85] M. A. Wolfe. Interval mathematics, algebraic equations and optimization. *J. Comput. Appl. Math.*, 124 :263–280, December 2000. Cité page 191.
- [86] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. McLachlan, A. Ng, B. Liu, P. Yu, Z. Zhou, M. Steinbach, D. Hand, and D. Steinberg. Top 10 algorithms in data mining. *Knowl. Inf. Syst.*, 14(1) :1–37, December 2007. Cité page 78.
- [87] B. Zhang, M. Hsu, and U. Dayal. K-harmonic means - a data clustering algorithm. Technical report, Hewlett-Packard Labs, 1999. Cité page 96.
- [88] Q. Zhang and I. Couloigner. A new and efficient k-medoid algorithm for spatial clustering. In Osvaldo Gervasi, Marina Gavrilova, Vipin Kumar, Antonio Laganà, Heow Lee, Youngsong Mun, David Taniar, and Chih Tan, editors, *Computational Science and Its Applications - ICCSA 2005*, volume 3482 of *Lecture Notes in Computer Science*, pages 207–224. Springer Berlin / Heidelberg, 2005. Cité page 99.

Index

Notations

$\#E$	111
χ	24
$\mathcal{B}(\mathcal{E})$	25
$\mathcal{I}_{\mathcal{D},x,r}$	112
$\mathcal{I}_{\mathcal{D},T,q}$	120
$\mathcal{F}(\mathcal{D},\mathcal{C},\overline{\mathcal{C}})$	120
$\overline{\mathcal{C}}$	111
$\tau(B,\mathcal{C},\overline{\mathcal{C}})$	113
$\theta_{\mathcal{C}}(B)$	112

A

alterné	25
analyse en composante principale	10
arbre	87
atteint pour	118

B

base fixe	35
base totalement caractéristique	112
base totalement discriminante	113
bases d'un matroïde orienté	23

C

bien classé par notre méthode	125
caractérisation d'une classe	9
cas uniforme	22
circuit	30
classification	9

classification supervisée	78
clustering	78
cocircuit	28
mal classé par notre méthode	125
pas classé par notre méthode	125
caractérisation faible	113, 120
caractérisation forte	114, 120
chirotope	25

D

diamètre d'un ensemble	148
discrimination de classes	9
distance de Hamming	82
distance de Manhattan	85
distance entre deux matroïdes orientés	34
distance euclidienne	85
distance mathématique	34
distance minimale entre deux ensembles	148
données binaires	77
données nominales	78
données numériques	77

H

hyperplan d'un matroïde orienté	28
hyperplan de \mathbb{R}^d	24

K

k -means	94
k -medoids	97

L

landmark	7
landmarks anatomiques.....	7
semi-landmarks	8

O

orientation d'un simplexe.....	24
--------------------------------	----

P

partition	9
point moyen.....	86
position générale.....	22

R

rang.....	24
-----------	----

S

simplexe.....	24
---------------	----

T

taux de caractérisation.....	112, 115
taux de discriminabilité.....	113, 118

U

uplet.....	22
uplet totalement caractéristique.....	115
uplet totalement discriminant.....	118



Table des figures

1.1	Exemples de landmarks anatomiques relevés sur un crâne humain.	8
1.2	Configurations de 5 points en 2 dimensions. Seul le point 5 a été déplacé.	10
1.3	Répartition des trois configurations de la Figure 1.2 en fonction de l'aire et de l'angle $\widehat{453}$	12
1.4	Configurations de points de la Figure 1.2 après l'étape de superposition de Procruste	13
1.5	Résultat de l'Analyse en Composante Principale	14
1.6	Déformations des configurations de points de la Figure 1.2.	15
1.7	Comparaison de deux crânes dont l'angle basicranial diffère	19
2.1	Exemple de 5 points en 3 dimensions	23
2.2	Les deux orientations possibles pour le tétraèdre ABCD	25
2.3	Configurations de 5 points en 3 dimensions	27
3.1	Ten anatomic points on a skull model [8]	43
3.2	View from the front	43
3.3	View from the right	43
3.4	View from the front	44
3.5	View from the left	44
3.6	View from below	44
3.7	G minimal with one vertex having degree 4	59
3.8	G minimal with two vertices having degree 3	59
3.9	O_2 is $<_x$	60
3.10	O_1 is $<_x$	60
3.11	$\{T_2, T_3\} = \{ACD, BCD\}$	61
3.12	$\{T_2, T_3\} \neq \{ACD, BCD\}$	61
4.1	Exemple d'une grille $G_{6,7}$ et de deux chemins correspondant à deux multi-ensembles	67
5.1	Exemple d'arbre	88

5.2	Arbre hiérarchique correspondant à la hiérarchie de l'exemple	89
5.3	Arbre hiérarchique représentant la hiérarchie indexée de l'exemple	89
5.4	Arbre hiérarchique correspondant à la hiérarchie de l'exemple	90
5.5	Arbres hiérarchiques en fonction de la distance utilisée	92
5.6	Comparaison entre l'algorithme des k -means et l'agorithme [54] basé sur les k -medoids	100
5.7	Exemple de situations où le résultat des k -medoids est moins bon que le résultat des k -means	101
5.8	Exemple d'un arbre de décision créé à partir de points en 2 dimensions	104
5.9	Graphique représentant le nombre d'erreurs de classification en fonction du nombre de feuilles	106
8.1	Position des 10 landmarks	142
8.2	Représentation des ordres sur une vue de face	142
8.3	Représentation des ordres sur une vue de profil	142
8.4	Schéma de la méthodologie	149
9.1	Position des 16 landmarks (image provenant de [8])	151
9.2	Moyenne des distances entre les chirotopes en fonction des classes définies par les experts	152
9.3	Moyenne des différences entre les espèces en fonction de l'âge	153
9.4	Représentation de la base ayant le plus fort taux de discriminabilité entre les humains et les singes	156
9.5	Vue de profil de "l'humain moyen"	156
9.6	Vue de profil de l'humain n'ayant pas pour B_1 la même orientation que la partie gauche de la Figure 9.4	156
9.7	Vue de profil du "singe moyen"	156
10.1	Représentation des classes d'Angle	159
10.2	Position des 10 landmarks	160
10.3	Moyenne des distances entre les chirotopes en fonction des classes	161
10.4	Une des quatre bases présentant le plus haut taux de discriminabilité	162
10.5	Une des quatre bases présentant le plus haut taux de discriminabilité	163
10.6	Exemple d'un modèle numérique tridimensionnel d'un crâne présentant de lourdes déformations crâniennes	164
11.1	Photographies d'un enfant présentant une craniosynostose coronale	165
11.2	Vue latérale du crâne dans laquelle la suture coronale est représentée en rouge	166
11.3	Illustration des 133 landmarks tridimensionnels	167
11.4	Moyenne des distances entre les chirotopes en fonction des classes	167
11.5	Matrice des distances entre les chirotopes	169
11.6	Moyenne des distances entre les chirotopes en fonction des classes	171
11.7	Matrice des distances entre les chirotopes	172
11.8	Une des quatre bases totalement discriminantes pour les RUCS et la base obtenue par symétrie qui est totalement discriminante pour les LUCS	174
11.9	Représentation des deux bases nous donnant une caractérisation des crânes BCS	175

11.10	Représentation des bases telle que l'orientation des deux bases soit $-$, ce qui correspond à une caractérisation des crânes BCS	175
11.11	Les autres orientations possibles correspondant à des crânes n'appartenant pas à la classe BCS	175
11.12	Position des 120 demi-crânes sur les deux premiers axes de l'Analyse en Composantes Principales réalisée par Heuzé et al. [31].	177
11.13	Tableau des distances entre toutes les paires de demi-crânes.	180
11.14	Deux exemples de quatre landmarks formant une base totalement discriminante entre le cluster {L-BCS, R-BCS} et les trois autres clusters.	184
11.15	Représentation d'une base totalement discriminante pour les demi-crânes des BCS (L-BCS et R-BCS)	185

Abstract

In this thesis, we propose an innovative combinatorial method based on oriented matroids for the quantitative study of the shape of 3D anatomical structures. We rely on landmarks which were previously defined by experts on the studied anatomical structure. The novelty of this method results from the use of oriented matroids. These mathematical tools allow us to encode the relative position of landmarks in a purely combinatorial way, that is without using concepts of angles or distances, by associating a sign (0, + or -) for each subset of $(d+1)$ landmarks where d is the dimension of space (in our case 2 or 3). In the first part, we assume that there exist constraints of orders on each coordinate axis for the landmarks. We obtain a characterization (in dimension 2 and 3) of the subsets of landmarks of which the associated sign is constant, regardless of the values of the coordinates satisfying the constraints of order. In a second part, we try to classify a set of 3D models, encoding in advance by these lists of signs. We first analyze how to apply classic clustering algorithms, and then describe how to characterize the classes directly, using signs associated with some subsets of landmarks. In the third part, we explain the algorithms and the implementation of this new morphometry method in order to apply it to real data. In the last part, we apply the method to three databases each consisting of several dozens of points defined on several dozens to several hundreds of cranial structures for applications in comparative anatomy, in orthodontics and on clinical cases of children with craniofacial deformities.

Keywords: *oriented matroids, 3D morphology, combinatorics, 3D modeling*

Résumé

Dans cette thèse, nous proposons une approche combinatoire novatrice fondée sur les matroïdes orientés pour l'étude quantitative de la forme de structures anatomiques 3D. Nous nous basons sur des points de repère qui ont été préalablement localisés par des experts sur la structure anatomique étudiée. La nouveauté de cette méthode provient de l'utilisation de matroïdes orientés. Ces outils mathématiques nous permettent de coder la position relative des points de repère de façon purement combinatoire, c'est-à-dire sans utiliser de notions d'angles ou de distances, en associant un signe (0, + ou -) à chaque sous-ensemble de $(d+1)$ points de repère où d est la dimension de l'espace (dans notre cas 2 ou 3). Dans une première partie, nous supposons qu'il existe des contraintes d'ordres sur chaque axe de coordonnées pour les points de repère. Nous obtenons alors une caractérisation (en dimension 2 et 3) des sous-ensembles de points de repère dont le signe associé est constant, quelles que soient les valeurs des coordonnées satisfaisant les contraintes d'ordre. Dans une deuxième partie, nous cherchons à classifier un ensemble de modèles 3D, en les codant au préalable par ces listes de signes. Nous analysons d'abord comment s'appliquent les algorithmes de clustering classiques, puis nous décrivons comment caractériser des classes de façon directe, à l'aide des signes associés à quelques sous-ensembles de points de repère. Dans une troisième partie, nous détaillons les algorithmes et l'implémentation en machine de cette nouvelle méthode de morphométrie afin de pouvoir l'appliquer à des données réelles. Dans la dernière partie, nous appliquons la méthode sur trois bases de données composées chacune de plusieurs dizaines de points de repères relevés sur plusieurs dizaines à plusieurs centaines de structures crâniennes pour des applications en anatomie comparée, en orthodontie et sur des cas cliniques d'enfants présentant des déformations cranio-faciales.

Mots clefs : *matroïdes orientés, morphologie 3D, combinatoire, modélisation 3D*
