École doctorale Sciences pour l'Ingénieur de l'Université de Lille

Thèse de Doctorat

Discipline

Mathématiques appliquées

présentée par

Hiba Alawieh

Fitting distances and dimension reduction methods with applications

Soutenue publiquement le 13 mars 2017 devant le jury composé de

Directeur de thèse:	Pr. Nicolas Wicker	Université de Lille 1, France
Co-directrice de thèse:	Pr. Baydaa AL Ayoubi	Université Libanaise, Liban
Rapporteurs:	Pr. Gérard Biau	UPMC, France
	Pr. Avner Bar-Hen	CNAM Paris, France
Examinateurs:	Pr. Gilbert Saporta	CNAM Paris, France
	Pr. Sophie Dabo	Univérsité de Lille 3, France

Contents

To my parents
who are my first mathematics teachers and so much more.
To my sisters
Aya, Maya and Rana ♡

Acknowledgments

First and above all, all thanks to almighty God, Allah, for the majestic grace, courage, and patience that has immensely guided me in finishing this thesis.

I would like to express my sincere appreciation to my thesis advisor Nicolas Wicker. His wide knowledge and experience have been of great value for me. His encouraging and personal guidance have provided a good background for the present thesis. I would thank him also for his patience, kindness, motivation and invaluable support during these years.

I also want to thank my co-supervisor Baydaa Al Ayoubi, for all of the advices and cooperation she gave during the years I spent in the Lebanese university and Lille 1 university.

Many thanks to professor Gérard Biau at Pierre et Marie Curie university and professor Avner Bar-Hen at CNAM Paris, for accepting to review my work as well as for participating in my jury committee.

I am also grateful to professor Gilbert Saporta at CNAM Paris and professor Sophie Dabo at Lille university for their commitment to take part in my jury committee.

I am particularly grateful to Myriam Maumy-Bertrand and Frédéric Bertrand at Strasbourg university for their warm hospitality and the time they devoted to me in a part of my work. Thank you for your kind collaboration.

A particular thank to professor Christophe Biernacki at Lille university, for our collaboration on a part of this thesis.

I will be always grateful to all of my laboratory partners for the good moments that we shared.

My gratitude goes to my friends for their support and for all the times we have spent together during these years.

Finally, I warmly thank and appreciate my parents who have always supported and helped me in life. Thank you for your endless love, encouragement and emotional support. I also would like to thank my sisters for the assistance they have provided in numerous ways.

Résumé

Les données observées dans les différentes études dans tous les disciplines de la science peuvent être mesurées selon deux dimensions, le nombre de variables et le nombre d'exemples et dans certains cas comme des distances ou dissimilarités entre les individus. Dans la plupart de ces études, le nombre de variables peut prendre des valeurs élevées ce qui rend leur analyse et leur visualisation assez difficile. Cependant, plusieurs méthodes statistiques ont été conçues pour réduire la complexité de ces données, en utilisant les coordonnées des individus ou bien les distances entre les individus, et permettant ainsi une meilleure compréhension des connaissances disponibles dans ces données.

Dans cette thèse, notre objectif est de proposer deux nouvelles méthodes d'analyse des données multivariées basé sur l'utilisation des distances entre les paires d'individu. Ces deux méthodes s'appellent en anglais : " *Multidimensional Fitting*" et "*Projection under pairwise distance control*".

La première méthode est une dérivée de la méthode de positionnement multidimensionnelle (*multidimensional scaling (MDS) en anglais*) dont l'application nécessite la disponibilité des deux matrices décrivant la même population : une matrice de coordonnées et une matrice de distances et l'objective est de modifier la matrice des coordonnées de telle sorte que les distances calculées sur cette matrice soient les plus proches possible des distances observées sur la matrice de distances. Nous avons élargi deux extensions de cette méthode : la première en pénalisant les vecteurs de modification des coordonnées et la deuxième en prenant en compte les effets aléatoires qui peuvent intervenir lors de la modification. Deux applications de ces deux extensions ont été faites sur des données biologiques et des données de la sensométrie.

La deuxième méthode est une nouvelle méthode de réduction de dimension basée sur la projection non linéaire des données dans un espace de dimension réduite et qui tient en compte la qualité de chaque point projeté pris individuellement dans l'espace réduit. La projection des points s'effectue en introduisant des variables supplémentaires, qui s'appellent "rayons", et indiquent dans quelle mesure la projection d'un point donné est

précise. Les principales contributions de cette méthode sont de donner une simple visualisation des données en $\mathbb{R}^2$ avec une interprétation simple de la qualité d'approximation et de fournir une nouvelle variante de réduction de la dimensionnalité. Nous avons appliqué cette méthode sur différents types de données tels que des données quantitatives, qualitatives et fonctionnelles.

Abstract

Data observed in various studies in all sciences disciplines can be measured in two dimensions, the number of variables and the number of examples and in some cases as distances or dissimilarity between points. In most of these studies, the number of variables can take high values which makes their analysis and visualization quite difficult. However, several statistical methods have been developed, using the coordinates of the points or the pairwise distance to reduce the complexity of these data, allowing a better comprehension of the knowledge available in these data.

In this thesis, our aim is to propose two new methods of multivariate data analysis based on the use of the pairwise distance. These two methods called: " *Multidimensional Fitting*" and "*Projection under pairwise distance control*".

The first method is a derivative of multidimensional scaling method (MDS) whose the application requires the availability of two matrices describing the same population: a coordinate matrix and a distance matrix and the objective is to modify the coordinate matrix such that the distances calculated on the modified matrix are as close as possible to the distances observed on the distance matrix. Two extensions of this method have been extended: the first by penalizing the modification vectors of the coordinates and the second by taking into account the random effects that may occur during the modification. Two applications of these two extensions have been done on biological data and data of sensometrics domain.

The second method is a new method of dimensionality reduction techniques based on the non-linearly projection of the points in a reduced space by taking into account the projection quality of each projected point taken individually in the reduced space. The projection of the points is done by introducing additional variables, called "radii", and indicate to which extent the projection of each point is accurate. The main contributions of this method are to give a simple data visualization in $\mathbb{R}^2$ with a straightforward interpretation of the approximation quality and provide a new variant of dimensionality reduction. We have applied this method on different types of data as quantitative, qualitative and functional data.

Publications

- Hiba Alawieh, Frédéric Bertrand, Myriam Maumy-Bertrand, Nicolas Wicker and Baydaa Al Ayoubi, *A random model for Multidimensional Fitting*. Submitted in computational statistics and data analysis.
- Hiba Alawieh, Nicolas Wicker and Christophe Biernacki. *Projection under pairwise distance control*. 2016. Submitted to Computational Statistics (COST), in revision.
- Hiba Alawieh, Nicolas Wicker, Baydaa Al-Ayoubi, and Luc Moulinier. *Penalized Multidimensional Fitting for Protein Movement Detection*. Journal of Applied Statistics(2016), DOI: 10.1080/02664763.2016.1261811.

List of Figures

List of Tables

Introduction

In many disciplines, researchers measure several variables on each individual or object. The use of univariate procedures with these data is not always possible. Elementary statistical analysis can be applied just to situations where one or two variables are observed on a set of statistical objects (populations or samples). Extending these methods to cases where the number of variables becomes higher is referred to multivariate data analysis.

Recent decades have seen impressive progress in multivariate data analysis and becomes increasingly popular among scientists and their applications in different fields such as biology, chemistry, physics, geology, psychology and many other fields. This enormous growth is a result of the development of computer science and statistical software.

Multivariate data analysis, presented in multidimensional space, studies the situations where a set of variables must be simultaneously studied. It attempts to provide results by reducing the number of dimension, and not limiting to one.

However, several statistical methods have been designed to reduce the complexity of these data by conserving as much as possible the information given by the initial data set. Different categories of multivariate data analysis have been developed as dimensionality reduction, variable selection, cluster analysis and others. These categories are directly related to the main goal of the researcher: dimensionality reduction consists in summarizing the data, variable selection consist in choosing the pertinent variables from the set of candidate variables and cluster analysis seeks grouping of the object or variables. Moreover, data visualization in small dimension space is a main topic in multivariate data analysis in many research domains so that many information can be given by this representation and gives then an overview of the information contained in the data. In this thesis, we focus our discussion on data dimensionality reduction and data visualization approaches which is at the centre of interest of in our thesis.

A long list of well-known methods are used to reduce the dimension of data or/and to visualize this data in reduced space. Principal components analysis (PCA) is the most popular multivariate data analysis method used to reduce and visualize data. Despite the simplicity and the efficiency of this method it presents many disadvantages. The

main drawbacks in PCA is firstly that it can only perform linear combinations though non-linear transformation can reduce the dimension of data if these data are positioned on the surface of a Swiss roll for example. Secondly, PCA assumes that the directions with the largest variances contain the most information which is not always valid as in the linear discriminant analysis where minimum-variance variable permits classes to be well separated. Additionally, the choice of the correct number of principal components to keep can also induce an ambiguity in the application of PCA. Therefore, developing methods that reduce the dimension of data and/or provide a simple data visualization without taking into account the linearity separation or any specific construction of data, is the main motivation of this thesis.

We will hereafter discuss two new methods for multivariate data analysis. The first one is based on fitting distances and the second one is based on non-linear projection points under pairwise distance control.

We first consider that we have two matrices, one contains the coordinates of the points and the other contains the pairwise distance. A new method has been developed called '*multidimensional fitting*' which modifies the coordinates of the points in such a way that the distances calculated on the modified coordinates are as close as possible to the distances given by the second matrix. The modification of the coordinates depends on the problem at hand and in our work, we consider the modification as displacements of the points that can be more interpretable. A penalized version of this method is needed to avoid unnecessary modification, we address '*penalized multidimensional fitting*'. This will be the focus of our work in chapter 2 with an interesting application to molecular biology.

Neglecting the presence of random effects of multidimensional fitting model can affect the interpretation of the displacements significance. Therefore, a random model of multidimensional fitting exhibiting how significant displacements are is needed. This will be the focus of our work in chapter 3 with an application to sensometrics.

Then we consider that we have just one matrix that contains the pairwise distance. Using these distances, we propose a new non-linear projection method that takes into account the local projection quality. As in many dimensionality reduction and data visualization method, the quality of projection is a globally quality measure that takes only into account what happens globally, a local quality measure can be very interesting to indicate the projection quality of each projected point taken individually. In PCA, the local measure is evaluated by the squared cosine of angle between the principal space and the vector of the point. This measure is useful in cases of linear projection as happens in PCA but cannot be applied to the case of non-linear projection. This leads to the

development of a novel method that project the points in a reduced space and take into account the projection quality of each point taken individually. This method refereed to as '*projection under pairwise distance control*' performs a straightforward data visualization in $\mathbb{R}^2$ with a simple reading of the approximation quality and provides a novel variant of dimensionality reduction. This will be the subject of our work in chapter 4 with different applications using different kinds of data set.

This thesis is composed of four chapters:

In **chapter 1**, we present some background methods and results. We begin first with an introduction about multivariate data analysis, then discuss two different ways to treat this kind of data: data dimensionality reduction and data visualization. After a brief introduction where the main ideas of data dimensionality reduction and data visualization are illustrated, different methods for each way are given.

In **chapter 2**, we are interested in the problem of distances fitting. A new method of multivariate data analysis based on the modification of one matrix to make it fit to a second matrix is presented. This method has been applied in the molecular biological domain particularly to detect the movement of amino acids in a protein. The penalization of the displacements is important to obtain good information on which part of protein has moved.

In **chapter 3**, a random model of multidimensional fitting extends the model seen in Chapter 2. Moreover, we introduce a statistical test to assess how much a transformation is significative. Optimization and simulation are illustrated to obtain the displacement vectors of the points with an application on sensometrics in order to explain consumer preferences for a set of competitive products by their sensory profiles for these products.

In **chapter 4**, a new projection paradigm is presented to describe a non-linear projection method that takes into account the projection quality of each projected point in the reduced space, this quality being directly available in the same scale as this reduced space. More specifically, this novel method allows a straightforward visualization data in $\mathbb{R}^2$ with a simple reading of the approximation quality, and provides then a novel variant of dimensionality reduction which is illustrated in the application to different types of data sets.

Chapter 1

Preliminaries

In this chapter some concepts, background methods and results are presented.

We begin first with an introduction about multivariate data analysis, used when several measurements are made on each individual or object in one or more samples. Then, some methods are introduced to facilitate the interpretation of this kind of data. In this section, different dimension reduction and data visualization methods considered in this thesis are presented.

1.1 Introduction to multivariate data analysis

In several scientific domains, researchers measure several variables on each individual or object. This data can be viewed as a large matrix and extracting results from this type of matrix is often hard and complicate since the use of univariate data analysis procedures with this data is not allowed. That's why, a new context of data analysis becomes a priority in order to extract as possible as the information contained in these types of matrices. We address multivariate data analysis.

Multivariate data analysis [1, 2, 3] provides methods used to reduce the complexity of data by retaining as much as possible data information. In recent decades, long list of multivariate data analysis methods have been developed and have become increasingly popular among scientists in all domains and their applications affecting various areas. In this chapter, we are interested by the dimensionality reduction methods called too projection methods.

Dimensionality reduction techniques can be used in different ways including:

- Data dimensionality reduction: project the data in the high-dimensional space to a low-dimensional space. Linear and non-linear data dimensionality reduction are distinguished.

- Data visualization: provide a simple interpretation of the given data in $\mathbb{R}^2$ or $\mathbb{R}^3$

Since our work is based on dimensionality reduction, two sections are presented to give an overview of existing methods from literature used in the data dimension reduction and data visualization.

1.2 Data dimensionality reduction

A variety of data dimensionality reduction methods have been proposed to drop the difficulties associated to the high dimensional data. These methods are divided into two categories: probabilistic and non-probabilistic methods. Firstly, we present a brief introduction of data dimensionality reduction and then we explore the non-probabilistic methods by discussing both linear and non-linear dimensionality reduction problems. After that, some well-known methods of data dimensionality reduction are given.

1.2.1 Introduction

Given n p-dimensional data points noted $X = [X_1|\dots|X_n] \in \mathbb{R}^{n\times p}$ and a choice of dimensionality $q < p$. The task of data dimensionality reduction is to find another data representation in dimension q noted $Y = [Y_1|\dots|Y_n] \in \mathbb{R}^{n\times q}$ which retains, at maximum, the same data information as the original data given by the matrix X.

Linear dimensionality reduction is performed to produce a linear transformation $P \in \mathbb{R}^{q\times p}$ of the form $Y = XP^{\mathrm{T}}$ and non-linear dimensionality reduction transforms the original data into a non-linear subspace using a non-linear function f.
Large number of non-probabilistic dimensionality reduction methods have been studied over the last decade to find appropriate linear and non-linear transformations. The state-of-the-art here is considered as a brief description of some well-known methods used to reduce dimensionality of data.

1.2.2 Non-probabilistic dimensionality reduction methods

In this section, we want to present some of the important methods used in the linear and non-linear dimensionality reduction. Principal component analysis (PCA), multidimensional scaling (MDS), Procrustes analysis and kernel PCA methods are presented.

1.2.2.1 Principal component analysis

The most commonly applied method in linear dimensionality reduction and also the oldest multivariate technique is principal component analysis (PCA). It is the "mother" of the multidimensional data analysis [4] and has been investigated extensively [5, 6, 7, 8]. The aim of this method is to project the original data set, using a linear projector, in a reduced space by preserving as much of the variance from the original data set as possible.

PCA computes new variables called *principal components* which are obtained as linear combinations of the original variables.

Recall that the singular value decomposition of matrix X with rank equal to r is given as: $X = UDV^T$, where $U \in \mathcal{M}_{n\times r}(\mathbb{R})$, $V \in \mathcal{M}_{p\times r}(\mathbb{R})$ are respectively matrix of left and right singular vectors and $D \in \mathcal{M}_{r\times r}(\mathbb{R})$ is the diagonal matrix of singular values.

Let M a chosen metric, we note I the total inertia of the original points and it is given by:

$$I = \sum_{i=1}^{n} p_i \|X_i - g\|^2 \tag{1.1}$$

where g is the center of gravity of the points and p_i is the weight of point i such that $\sum_{i=1}^{n} p_i = 1$. This inertia can be rewritten as:

$$I = \text{Trace}(MV). \tag{1.2}$$

We have defined $P \in \mathcal{M}_{q\times p}(\mathbb{R})$ as a linear transformation which is called in PCA the projection operator. The PCA comes down to find the projection operator by maximizing the inertia of the projected data.
Indeed, the projected data Y is given by: $Y = XP^{\mathrm{T}}$ and the covariance matrix of Y is given by PVP^{T} that implies that the inertia of projected points is equal to $\text{Trace}(PVP^{\mathrm{T}}M)$ and by elementary operations, we conclude that $\text{Trace}(PVP^{\mathrm{T}}M) = \text{Trace}(VMP)$. So, maximizing the inertia of the projected data (*i.e.* $\text{Trace}(VMP)$) leads to conclude that the q principal components are obtained from the q eigenvectors of the matrix VM and explained the largest part of the inertia of the original data.

1.2.2.2 Multidimensional Scaling

Likewise PCA, Multidimensional scaling (MDS) consists in finding a new data configuration in a reduced space. The main difference between these two methods is that the input data in MDS are typically comprehensive measures of similarity or dissimilarity between objects, they are called "proximity". These proximities are arranged in a $n \times n$ symmetrical matrix called "proximity matrix" noted D and having d_{ij} as general element where d_{ij} is the proximity between i and j. The key idea of MDS is to perform dimensionality reduction in a way to approximate high-dimensional distances noted d_{ij} by the low-dimensional distances δ_{ij}.
Metric and *nonmetric* MDS can be shown by distinguishing the type of original distances. Metric MDS [9] is used when the original data are in the form of observed quantities measured in terms of coordinates where distance is meaningful. Nonmetric MDS is a rank-based approach which means that the original distance data is substituted with ranks ([10], [11] and for a recent thorough review see [12]). It is resorted to when the original data are of categorical or of a similar type where only ranking is important and not actual differences.

The objective function of metric MDS minimization problem noted *Stress* is the following:

$$Stress = \frac{\sum_{1\leq i<j\leq n}(d_{ij} - \delta_{ij})^2}{\sum_{1\leq i<j\leq n} d_{ij}}, \tag{1.3}$$

and those of nonmetric MDS is:

$$Stress = \frac{\sum_{1\leq i<j\leq n}(f(d_{ij}) - \delta_{ij})^2}{\sum_{1\leq i<j\leq n} f(d_{ij})}. \tag{1.4}$$

The function f that appears in the nonmetric MDS is a monotonically increasing function chosen always to minimize *Stress* and works as if it was a regression curve with $f(d_{ij})$ as y and δ_{ij} as $\hat{y}$.

In what follows, we are interested in the presentation of the basic steps of the metric MDS method. The first step of this method is to construct the matrix $B = \{b_{ij}\}$ such that

$$b_{ij} = -\frac{1}{2}\left(a_{ij} - a_{i.} - a_{.j} + a_{..}\right), \tag{1.5}$$

where

$$a_{ij} = d_{ij}^2, a_{i.} = \frac{1}{n}\sum_{a=1}^{n} d_{ia}^2, a_{.j} = \frac{1}{n}\sum_{a=1}^{n} d_{aj}^2 \text{ and } a_{..} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}^2$$

The optimal value of the objective function is computed from the spectral decomposition of B. Let V denote the matrix formed with the first q eigenvectors and Λ is a diagonal matrix containing the q eigenvalues associated to the q chosen eigenvectors as diagonal terms. The projected data points in the lower dimensional space are then given by:

$$[Y_1|, \ldots, |Y_n] = \sqrt{\Lambda}V^T. \tag{1.6}$$

Note that since Euclidean distances are used, the projected data points in the lower dimensional space are equivalent to those obtained using PCA [16].

For more details about nonmetric MDS and the different steps of resolution see [11], [12], [13], [14] and [15].

Moreover, MDS has been recently extended to perform non-linear dimensionality reduction. A recent approach of non-linear dimensionality reduction based on MDS is the Isomap algorithm. More details about Isomap can be founded in [20] and [21].

1.2.2.3 Procrustes analysis

Procrustes analysis [17] takes his name from the Procrustes torture method from Greek mythology how fit the victims to his bed. It is a goodness of fit method based on the transformation of one configuration to map another configuration as closely as possible through rotation, translation and scaling [2].

Let us consider two configurations $X_1, \ldots, X_n$ and $X'_1, \ldots, X'_n$, the idea is to modify $X'_1, \ldots, X'_n$ by translating and rotating it through a vector b and an orthonormal matrix A respectively. So, the problem can be written as the following minimization problem:

$$E_{PA} = \sum_{i=1}^{n} \|X_i - AX'_i - b\|^2. \tag{1.7}$$

The vector b is given by a simple derivation of E_{PA} with respect to b as follows:

$$b = \frac{1}{n} \sum_{i=1}^{n} (X_i - AX'_i). \tag{1.8}$$

The value of b is equal to zero if the two configurations are centered.

To find the matrix A, we suppose that the two configurations are centered and we note $Z = X'^T X$. By supposing Z of rank equal to p, the singular value decomposition of matrix Z is given by: $Z = UDV^T$. By taking into account these suppositions and using a matrix derivation with respect to A we obtain:

$$A = VU^T. \tag{1.9}$$

A scale change is also possible, the only modification is to take cAX'_i instead of AX'_i with c is a scale change parameter.

More generally, Generalized Procrustes analysis (GPA) consists in considering m configuration instead of two. We note consensual configuration the average configuration calculated from the m configurations. GPA is an iterative method that reduces by a series of transformations (scaling, translations, rotations, reflections) for the m configurations, the distance of the m configurations to the consensus configuration. We refer the reader to [18] for a detailed analysis of GPA.

1.2.2.4 Kernel PCA

In recent years, PCA has been extended to work with non-linear configurations as classical PCA will fail to give us a "good" representation. Kernel PCA [19] has been developed to extract non-linear principal components without expensive computations.

The idea is to perform PCA in a feature space noted $\mathcal{F}$ produced by a non-linear mapping of data from its space into the feature space $\mathcal{F}$, where the low-dimensional latent structure is, hopefully, easier to discover. The mapping function noted Φ is considered as:

$$\begin{array}{rccc} \Phi: & \mathbb{R}^p & \rightarrow & \mathcal{F} \\ & X & \rightarrow & \Phi(X) \end{array}$$

The original data X_i is then represented in the feature space as $\Phi(X_i)$, where $\mathcal{F}$ and Φ are not known explicitly but obtained thanks to the "kernel trick". We assume that we are dealing with centered data i.e. $\sum_{i=1}^{n} \Phi(X_i) = 0$.

The main steps of kernel PCA are the following:

- Compute the Gram matrix $K = k_{ij}$ with $k_{ij} = K(X_i, X_j) = \langle \Phi(X_i), \Phi(X_j) \rangle$ is a chosen kernel.

- Compute the centred Gram:

$$\tilde{G} = HKH. \tag{1.10}$$

- Find the eigenvalues λ_i and the eigenvectors V_i of matrix $\tilde{G}$ satisfying

$$\lambda V = n\tilde{G}V. \tag{1.11}$$

- The eigenvectors are then expressed as:

$$V_l = \sum_{i=1}^{n} \alpha_i^l \Phi(X_i), \tag{1.12}$$

where α_i^l is calculated from the equation:

$$\lambda_i n \alpha_i^l \alpha_i^{l^T} = 1. \tag{1.13}$$

These eigenvectors can be used as the eigenvectors of PCA.

1.3 Multidimensional data visualization

Another objective in dimensionality reduction is to visualize with the naked eye the information contents in the data. It is natural to show that a simple data visualization is given with a small projection space dimension (i.e. $k = 2$ or 3). Multivariate data visualization refers to the display of multidimensional data and aids the user to gain insight into the data and directly interact with the data.
Different Multidimensional data visualization techniques have been developed and divided into different categories [22], [23]. Four main categories are distinguished as geometric projection techniques, Pixel-oriented techniques, hierarchical display techniques, iconography techniques. In the following, we present briefly some of these methods.

1.3.1 Data Visualization techniques

This section is concerned to present some multidimensional data visualization techniques. Five methods are detailed, namely thye Scatter plot Matrix, Parallel coordinates, Self-Organizing Maps (SOM), PCA and Sammon's mapping.

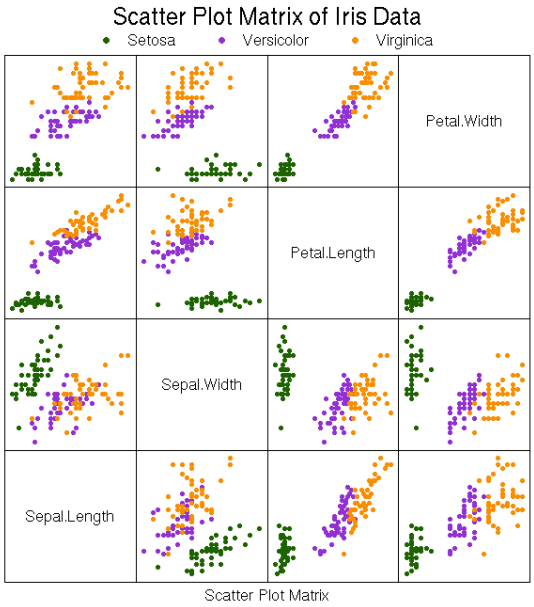


Figure 1.1: Scatter plot matrix of Iris data set.

1.3.1.1 Scatter plot Matrix

Scatter plot is a geometric visualization method [25] based on all the possible two dimensional data plots. The scatter plot matrix is composed of $\frac{p(p-1)}{2}$ scatter plots of all possible pairs of variables in a dataset and give the relationships between pairs of several variables [24].

An example of scatter plot method is given in figure 1.1 which depicts the scatter plot matrix of the well-known Irises data set [29]. We can see that Setosa flowers are remote from Versicolor and Virginica.

1.3.1.2 Parallel coordinates

Parallel coordinates method is also a geometric visualization method introduced by Alfred Inselberg [26]. In this method, variables are represented as parallel lines such that the upper and the lower points of each line are obtained respectively by the maximum and the minimum of each variables. A n-dimensional point is displayed as $n-1$ line segments that cross each axis at a position proportional to its value for that dimension.

The Irises data set is displayed by applying parallel coordinates in figure 1.2. Each iris is represented as line segment with specific color corresponding to the three species of irises (blue correspond to Setosa, red to Versicolor and green to Virginica). It is clear to see that the species are distinguished best by the petal length and width than the sepal length and width.

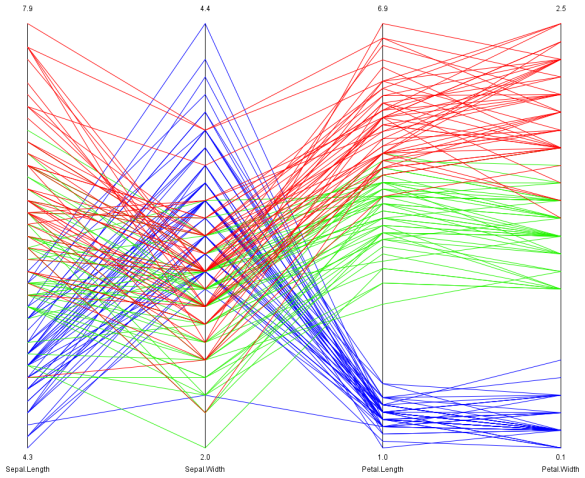


Figure 1.2: Parallel coordinates of Iris data.

1.3.1.3 Self-Organizing Maps (SOM)

The Self-Organizing Map [27] was developed by Kohonen in the early 1980's. It is one of the most popular neural network models based on unsupervised learning. The principal goal of SOM is to transform an incoming signal pattern of arbitrary dimension into a one or two dimensional discrete map, by making an artificial topographic maps that preserves neighbourhood relations. Indeed, each point in the output space will map to a corresponding point in the input space such that if X_1 and X_2 are two input vectors and Y_1 and Y_2 are two output vectors corresponding to X_1 and X_2. Then Y_1 and Y_2 must be close in the discrete low- dimensional output space if X_1 and X_2 are close in the continuous high dimensional input space.

A way to visualize the data is to represent each data dimension using feature planes by displaying graphically the levels of the variables in each map unit. Figure 1.3 depicts the feature planes obtained using the Iris data set. The distance between the adjacent neurons is calculated and presented with different colourings between the adjacent nodes such that the colors white and yellow signify high values of the variables, and black correspond to low values of the variables.

In Figure 1.3, U-matrix representation of SOM which visualizes the distance between the neurons shows that the top three rows form a cluster and by referring to the labels, we can see that this cluster corresponds to the specie Setosa whereas the other two species are not well separated. Moreover, petal length and petal width are very closely related to each other and also having some correlation with the sepal length shown that have similar component planes. Virginica specie is distinguished from Versicolor specie by their bigger leaves whereas Setosa is distinguished from them by the small and short petals and the wide sepals.

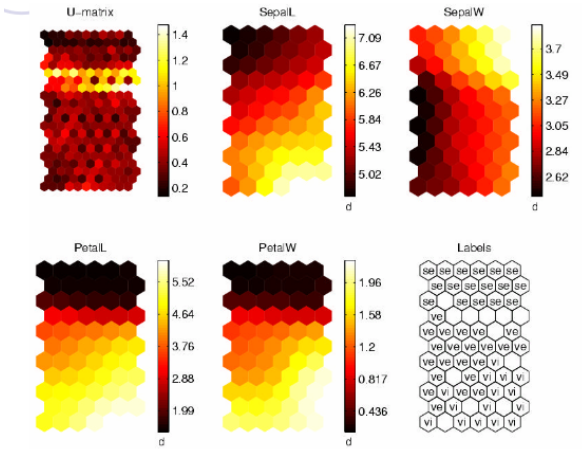


Figure 1.3: Feature planes created with SOM of Iris data set.

1.3.1.4 PCA

PCA presented in section 1.2.2.1 is not just useful as a data reduction method but it can serve also as a visualization method by taking $k = 2$. Figure 1.4 shows PCA plot that was constructed from the Iris data set.

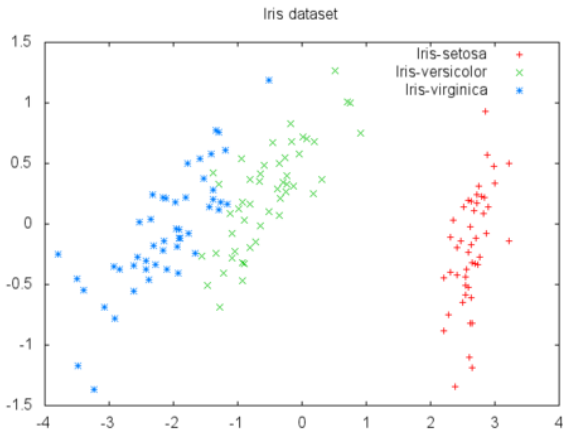


Figure 1.4: PCA of Iris data set.

The amount of variation explained by the two components is 73% + 22.9%= 95.9% of the total variance. We can see clearly and straightforwardly the separation of the specie Setosa from Versicolor and Virginica species.

1.3.1.5 Sammon's mapping

Sammon's mapping [28] is a non-linear projection method used in data visualization to an implicit data transformation by conserving as possible as the inter-point Euclidean

distances in the two spaces. It is based on the construction of a new lower-dimensional data set which has a similar structure to the first data set in order to reflects the structure present in the original dataset.

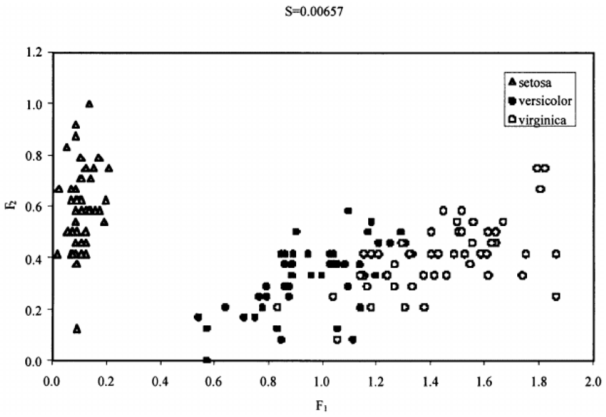


Figure 1.5: Sammon's mapping of Iris data set.

A two-dimensional visualisation of the structure of a four-dimensional of Iris data set is produced using Sammon's mapping and shown in figure 1.5. The value of the stress of the mapping obtained by the formula:

$$Stress = \frac{1}{\sum_{1\leq i<j\leq n}\delta_{ij}} \sum_{1\leq i<j\leq n} \frac{(d_{ij}-\delta_{ij})^2}{\delta_{ij}}, \tag{1.14}$$

is equal to 0.00657 which is very low (stress « 0.1), that indicates that the mapping gives a good idea of the underlying structure of the data. Note that d_{ij} and δ_{ij} are respectively the distances between i and j in the high-dimensional and low-dimensional spaces.
In [28] it was mentioned that the search of a minimum of Stress function is obtained using steepest descend procedure [30] as the minimum of the Stress function can not be found analytically, therefore the use of an iterative method for obtaining an approximate solution is allowed.

Bibliography

[1] Anderson, T.W. An Introduction to Multivariate Statistical Analysis. Wiley-Interscience, New York, 2003.

[2] Mardia, K.V., Kent, J.T., and Bibby, J.M. Multivariate Analysis. Academic Press, New York, 1979.

[3] Muirhead, R.J. Aspects of Multivariate Statistical Theory. Wiley-Interscience, New York, 2005.

[4] Saporta, G. Probabilités, analyse des données et statistique, Éditions Technip, 2006.

[5] Jackson, J. A Users Guide to Principal Components. John Wiley & Sons, New York, 1991.

[6] Jeffers, J. Two case studies in the application of principal component analysis. Appl. Statist. 16, 225?236, 1967.

[7] Joliffe, I. Principal Component Analysis. Springer-Verlag, 1996.

[8] Rao, C. The use and interpretation of principal component analysis in applied research. Sankhya A 26, 329–358, 1964.

[9] Togerson, W.S.Theory and methods of scaling. New York: Wiley, 1958.

[10] Shepard, R.N. Metric structure in ordinal data. Journal of Mathematical Pyschology, 3, 287–315, 1966.

[11] Kruskal, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika, 29, 1-27, 1964.

[12] Borg, I. and Groenen, P. Modern multidimensional scaling. New York: Springer Verlag, 1997.

[13] Kruskal, J. B. and Wish, M. Multidimensional Scaling. Sage University Paper series on Quantitative Applications in the Social Sciences, number 07-011. Sage Publications, Newbury Park, CA, 1978.

[14] Kenkel, N.C. and Orloci, L. Applying Metric and Nonmetric Multidimensional Scaling to Ecological Studies: Some New Results. Ecology 67, 919–928, 1986.

[15] Kruskal, J. B. Nonmetric multidimensional scaling: A numerical method. Psychometrika, 29, 2, 115-129, 1964.

[16] Andrew, A. M. Statistical pattern recognition. Robotica, 18(2), 219–223, 2000.

[17] Gower, J.C. and Dijksterhuis, G B. Procrustes Problems, Oxford Statistical Science, series 30, 2004.

[18] Gower, JC. Generalized procrustes analysis. Psychometrika, 40, 33-51, 1975.

[19] Scholkopf, B., Smola, A. and Muller, K.R. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. Technical Report, 44, 1996.

[20] Tenenbaum, J. B., de Silva, V. and Langford, J. C. A global geometric framework for nonlinear dimensional reduction, Science, 290, 2000.

[21] Yang, M.H. Face recognition using extended isomap, In Proceeding of ICIP, 2002.

[22] Keim, D. A. and Kriegel, H. P. Visualization Techniques for Mining Large Databases: A Comparison. IEEE Transactions on Knowledge and Data Engineering, vol.8, no.6. pp. 923-938, 1996.

[23] Chan, W. W-Y. A survey on multivariate data visualization in Science and technology. Department of Computer Science and Engineering Hong Kong University of Science and Technology Clear Water Bay, Kowloon, Hong Kong, 2006.

[24] Brian, S. E. and David, C. H. Encyclopedia of Statistics in Behavioral Science. John Wiley & Sons, Ltd, Chichester, 2005.

[25] Cleveland, W.S., McGill, M.E. Dynamic Graphics for Statistics. Wadsworth and Brooks/Cole, Pacific Grove, CA, 1988.

[26] Inselberg, A., The Plane with Parallel Coordinates, Special Issue on Computational Geometry, The Visual Computer 1, 69–91, 1985.

[27] Kohonen, T., Self-Organizing Maps, Third edition, Berlin: Springer-Verlag, 2001.

[28] Sammon, J., A non-linear mapping for data structure analysis, IEEE Transactions on Computers,18, 5, 401–409, 1969.

[29] Anderson E., The Irises of the Gaspé Peninsula. Bull. Am. Iris Soc. 59: p. 2–5, 1935.

[30] Freund, R.M.,The Steepest Descent Algorithm for Unconstrained Optimization and a Bisection Line-search Method, Massachusetts Institute of Technology, 2004.

Chapter 2

Penalized Multidimensional Fitting

In this chapter we are interested in the problem of protein movement detection producing upon the reaction it undergoes and its substrate/cofactor/partners binding state. Various methods exist to study these conformational changes but are limited to the detection of large movements. Our main contribution is to develop a new multivariate data analysis method based on fitting distances and inspired from MDS to gives a simple way to show these conformational changes. Experimental results that illustrate protein movements, as well as performance comparisons with current practice, are presented.

2.1 Introduction

Proteins are heteropolymers that can take three dimensional structures. These structures are flexible, highly dynamic, and their biological functions depend intimately on them [28]. They can change their shape, unfold (and refold), and switch to another conformation. This structure deformation can be induced in various ways such as binding other molecules –ligand binding– [22], enzyme catalysis reaction [10], etc. Figure 2.1 shows the ligand-binding reaction occurred on FhuA protein, an outer membrane transporter for ferrichrome-iron.

Two main methods are used to study the conformational changes of proteins: molecular dynamics simulation [1] and normal mode analysis [26]. Both of these methods use chemico-physical properties to predict motions. Several other algorithms are applied for detecting dynamical domains in proteins [21, 29, 11, 12, 4] and most of these algorithms define a rigid domain as a region whose internal geometry is the same in both conformations. Another approach is provided by a method called DynDom which determines domain dynamics, hinge axes, and hinge-bending residues using two conformations of the same protein [23]. However, this method, based on clustering, and the other algorithms cited above are mainly used for studying domain movements, so the main disadvantage for these methods is that they detect parts of amino acids sequence without giving us

any information on the movements of each amino acids taken alone, thus the movements comparison of amino acids it not possible. Therefore, a complementary method is needed for detecting more subtle changes.

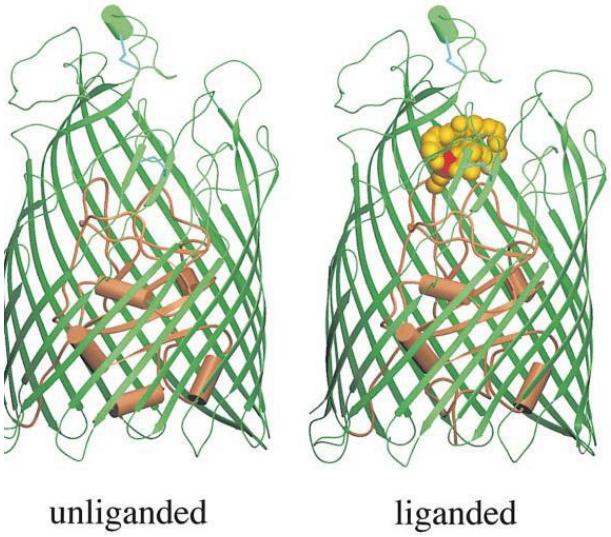


Figure 2.1: The liganded and unliganded forms of FhuA protein. The image is adapted from [16].

Here, a novel method called penalized Multidimensional Fitting (penalized MDF) is presented to detect movements by using two conformations of the same protein. This new multivariate analysis approach is an adaptation of Multidimensional Fitting (MDF) [3]. The idea is to compare one protein conformation with another one by modifying the coordinate matrix of the first one, called target matrix, in order to make the distances calculated on the modified target matrix similar to the distances given by the reference matrix that is obtained, in the case of protein movement detection, by computing the distances between its amino acids. The organization chart given in Figure 2.2 explains the procedure of this method:

It is composed of three steps:

- Step I: Extract the Cartesian coordinates for each conformation.
- Step II: Keep the coordinates of the first conformation in a matrix called **Target matrix** and calculate a distance matrix that contains the distances between the amino acids for the second conformation and called **Reference matrix**.
- Step III: Modify the target matrix in order to make the distances computed on this matrix after modification similar to the distances given by the reference matrix.

What differentiates our method from other goodness of fit measures [9],[8] used to compare two configurations like the Procrustes analysis [7] is that the latter transforms one

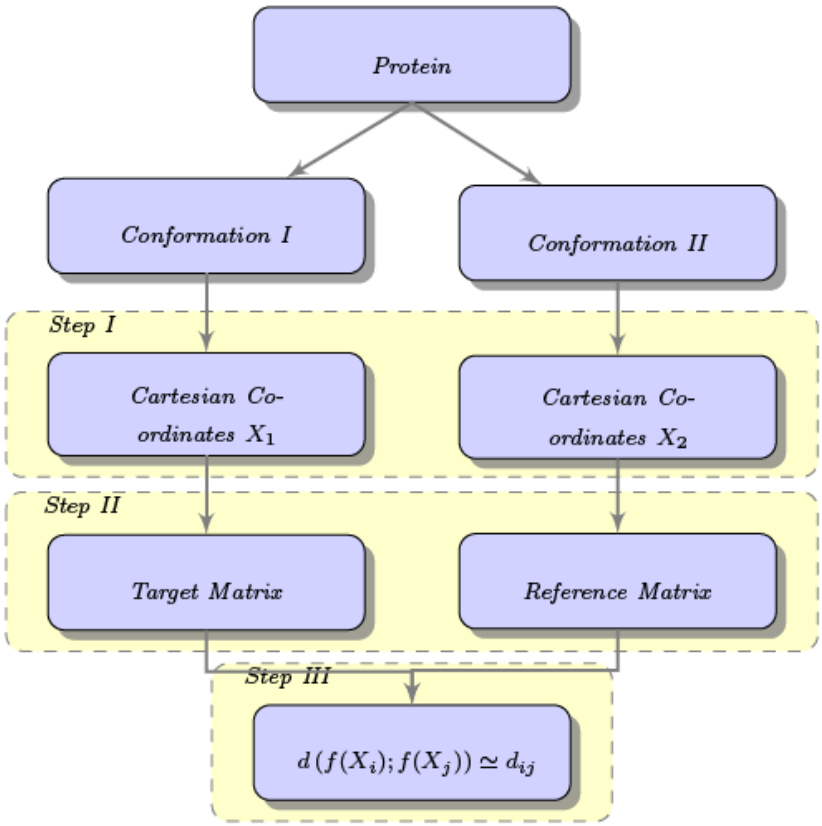


Figure 2.2: Penalized multidimensional fitting organization chart.

configuration to map another configuration as closely as possible through rotation, translation and scaling [17], that is the same rotation matrix, translation vector and scale are applied for all points.

Penalization is necessary as it is clear that without it, every transformation would be possible, and then the solution would reduce to take the modified target matrix equal to the coordinate matrix corresponding to the reference matrix! This of course, would not give any information on which part of the protein has moved. The main work here is to devise a good penalization. Once penalized MDF is applied, we compare the initial target matrix and the modified target matrix by computing the distance between the initial position of each amino acid and its final position. Important displacements indicate parts of the protein which have moved significantly between the two conformations.

In the second section, we have described the method. In the third section, it is applied on three different proteins. In the fourth section, we have compared our results to the results obtained by Procrustes analysis and DynDom. Finally, using k-means, we have simplified our optimization problem by transforming the reference matrix into a sparse matrix.

2.2 Penalized multidimensional fitting method

2.2.1 Introduction

Let $X = \{X_1|\cdots|X_n\}$ be the $n \times p$ target matrix given by the points values for the set of p variables and $D = \{d_{ij}\}$ the $n \times n$ reference matrix obtained separately from the pairwise dissimilarity (Table 2.1).

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} \quad D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ d_{n1} & d_{n1} & \cdots & d_{nn} \end{bmatrix}$$

Table 2.1: Target and reference matrix. X is the target matrix and D is the reference matrix

Multidimensional fitting (MDF) method presented for the first time in [3] allow us to modify the target matrix in order to minimize the difference between the reference matrix and the novel distance matrix computed on the modified target matrix. we note $\triangle = \{\delta_{ij}\}$ the distance matrix contained the pairwise distance after modification of target matrix X. The modification of coordinate X_i for $i = 1, \ldots, n$, is performed through an arbitrary function f as following:

$$\begin{array}{rccc} f: & \mathbb{R}^p \times \mathbb{R}^p & \longrightarrow & \mathbb{R}^p \\ & (X_i, L_i) & \longrightarrow & X_i + L_i. \end{array} \tag{2.1}$$

For all $i \in 1, \ldots, n$, the vector $L_i = (l_{i1}, l_{i2}, \ldots, l_{ip})$ denotes the displacement for the i^{th} point. The idea behind MDF is to minimize the mean square error:

$$\frac{2}{n(n-1)} \sum_{1 \le i < j \le n} (a\delta_{ij} - d_{ij})^2 ,$$

where $\delta_{ij} = d(f(X_i), f(X_j)) = d(X_i + L_i, X_j + L_j)$ under some constraints and $a \in \mathbb{R}$ is a scaling variable.

Here, no constraint is needed but to avoid unnecessary displacements, a penalty term is added to the mean square error leading to the following function to optimize:

$$E = \sum_{1 \le i < j \le n} (a\|X_i + L_i - X_j - L_j\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \text{pen}(L_i), \tag{2.2}$$

The optimization problem is then:

$$(P): \min_{L_1,\ldots,L_n \in \mathbb{R}^p} \sum_{1 \le i < j \le n} (\|X_i + L_i - X_j - L_j\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \text{pen}(L_i)\cdot$$

The parameter λ is a positive regularization parameter that controls the trade-off between the approximation of the reference matrix by the distance matrix computed on the modified matrix and the use of a parsimonious number of displacements. If $\lambda = 0$, the solution of (P) comes down to move all the points so that we obtain same matrices. When λ increases, more importance is given to the penalty term $\lambda \sum_{i=1}^{n} \text{pen}(L_i)$ eliminating unnecessary movements, then when λ becomes too large, $\text{pen}(L_i)$ tends to zero and nothing moves, which means that the initial values of the target matrix are kept. To have interesting results, it is clear that having a good penalization is important.

Returning to our protein movement detection problem, the application of penalized MDF is performed by constructing the target matrix X using the amino acids Cartesian coordinates of one protein structure and the reference matrix D using the Euclidean distances between the amino acids calculated on another structure of the same protein. So, the objective is to detect the amino acids that undergo an important movement by fitting the distances of one conformation to the distances of the second one by modifying only the coordinates of the first one. Here, the Cartesian coordinates of the two conformations are in the same scale so the scaling parameter a is taken equal to 1.
As next step, we have firstly discussed the choice of the penalty function and then the choice of the factor λ in the case where $a = 1$.

2.2.2 Choice of Penalty Function

Three natural possibilities for the choice of penalty function have studied such that ℓ_0, ℓ_1 and ℓ_2-norms.
First, we want to discard the ℓ_1-norm, to have a norm invariant by rotation as the structures can have any orientation. The ℓ_2-norm is used to penalize the displacements of points. This chosen norm can be used in many ways, we have only considered two cases, either having simply $\|.\|_2$ or having two homogeneous terms by taking $\|.\|_2^2$. Now if we consider two simple situations described in figure 2.3, we obtain the results described in table 2.2.

	$\sum_{i=1}^{4} \|l_i\|_2$	$\sum_{i=1}^{4} \|l_i\|_2^2$	$\sum_{i=1}^{4} (\|l_i\|_2 + \|l_i\|_0)$
case 1	$4d$	$4d^2$	$4d+4$
case 2	$4d$	$8d^2$	$4d+2$

Table 2.2: Penalty term values using different forms of penalty term in the cases 1 and 2.

Using $\|.\|_2^2$, we have obtained, with less points moving, a penalty larger to the one obtained with more points moving. This result is not interesting for our parsimony needs. Therefore, we have concentrated on norm $\|.\|_2$. Furthermore, to take into account the

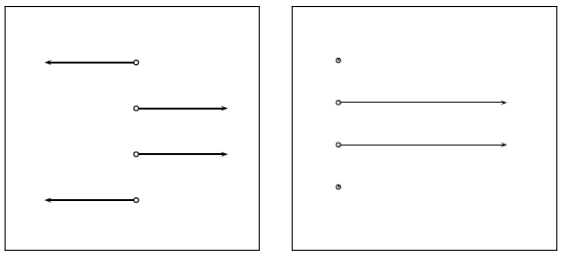


Figure 2.3: In case 1, two points move in the right direction and two others in the left one with the same absolute value of displacement that is equal to d. In case 2, only two points move but the final relative positions are the same between case 1 and 2.

number of points that move we have used the ℓ_0-norm. The combined penalty term $\sum_{i=1}^{4}(\|l_i\|_2 + \|l_i\|_0)$ give with less points moving, a penalty term smaller than that obtained with more points moving. So, we have used in our study this combined penalty term.
The ℓ_0-norm is not treated at a first stage to focus on the ℓ_2-norm. In this stage, the objective function is the following:

$$E = \sum_{1\leq i<j\leq n} \left(\|X_i + L_i - X_j - L_j\|_2 - d_{ij}\right)^2 + \lambda \sum_{i=1}^{n} \|L_i\|_2 \cdot$$

In a second stage, the results are transposed to the ℓ_0-norm in an elastic-net like context. The combined penalty term is $\sum_{i=1}^{n}(\gamma\|L_i\|_2 + (1-\gamma)\|L_i\|_0)$ weighted by a parameter λ, with $\gamma \in [0,1]$. We call the function $\gamma\|L_i\|_2 + (1-\gamma)\|L_i\|_0$ the elastic-net penalty by analogy with the well-known elastic-net [30]. The function to optimize becomes:

$$E = \sum_{1\leq i<j\leq n} \left(\|X_i + L_i - X_j - L_j\|_2 - d_{ij}\right)^2 + \lambda \sum_{i=1}^{n} \left(\gamma\|L_i\|_2 + (1-\gamma)\|L_i\|_0\right) \cdot$$

Note the two special cases $\gamma = 1$ and $\gamma = 0$. In the first case, the penalty term is reduced to the ℓ_2-norm, and in the second, to the $\ell_0$norm.

2.2.3 Choice of Parameter λ

We have already seen that the value of λ is crucial for obtaining good results. In this section, we want to find the best value of λ using ℓ_2-norm as penalty function (first stage). First, in Lemma 2.1 we show that if $\lambda > 0$ there are only fixed point or at least two points moving in different directions. Moreover, in the one dimensional case if there are no fixed points and for $\epsilon > 0$, we want to ensure that at least one $\|l_i\|_2$ exceeds ϵ then we derive in Lemma 2.2 that $\lambda < n(d_{ij}^0 - d_{ij} - 2\epsilon)$ with d_{ij}^0 is the initial distance computed on

the target matrix. Furthermore, Lemma 2.3 indicates that the optimum can be found in some specific cases provided that λ is inferior to $\frac{n-1}{2}\varepsilon$ where ε is the typical movement amplitude which is of the same order of magnitude as $(d^0_{ij} - d_{ij} - 2\epsilon)$ if we consider that $d^0_{ij} - d_{ij}$ is close to 2ϵ.

Lemma 2.1. *The solution of problem (P) is such that there is a fixed point or at least two points moving in different directions provided that $\lambda > 0$.*

Proof. We prove it by contradiction. Supposing that the optimal solution of (P) is L^o with all points moving in the same direction u, then we obtain:

$$E(L^o) = \sum_{1\leq i<j\leq n} (\|X_i + L^o_i - X_j - L^o_j\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \|L^o_i\|_2.$$

Now, let us consider a new solution $L^\dagger$ obtained from L^o by translating according to the smallest displacement of length m observed in L^o. We write: $m = \min_{i=1\ldots n} \|L^o_i\|_2$ and then the novel solution is given by: $\forall i = 1 \ldots n$, $L^\dagger_i = L^o_i - mu$. The objective function becomes:

$$\begin{aligned} E(L^\dagger) &= \sum_{1\leq i<j\leq n} (\|X_i + L^o_i - mu - X_j - L^o_j + mu\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \|L^o_i - mu\|_2 \\ &= \sum_{1\leq i<j\leq n} (\|X_i + L^o_i - X_j - L^o_j\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \|L^o_i - mu\|_2. \end{aligned}$$

Thus, $E(L^\dagger) < E(L^o)$ which is impossible by hypothesis. □

In Lemma 2.2, we prove that in one-dimensional case, if we want to have at least one $\|l_i\|_2 > \epsilon$, then λ cannot be too large. In Lemma 2.3, we prove, in dimension p, the same context of Lemma 2.2 that in a specific case, λ must not be too large.

Lemma 2.2. *In one dimension, at the optimum if for two successive points κ_1 and κ_2 ($\kappa_2 = \kappa_1 + 1$) having $x_{\kappa_1} < x_{\kappa_2}$ and moving such that $l_{\kappa_1} > 0$ and $l_{\kappa_2} < 0$; the parameter λ is such that $\lambda < n(d^0_{\kappa_1\kappa_2} - d_{\kappa_1\kappa_2} - 2\epsilon)$ and if for all $1 \leq i < j \leq n$, $x_i + l_i \leq x_j + l_j$, then $\exists\ k$ with $k \in \{1, \ldots, n\}$, such that $|l_k| > \epsilon$.*

We prove it by contrapositive assuming that $\forall i$ we have $|l_i| \leq \epsilon$. In one dimension, we note $l_1, \ldots, l_n$ the optimal displacement of points. Let:

- I^+ the set of all points that move in positive direction, we note: $I^+ = \{i | l_i > 0\}$.
- I^- the set of all points that move in negative direction, we note: $I^- = \{i | l_i < 0\}$.
- I^0 the set of all points that don't move, we note: $I^0 = \{i | l_i = 0\}$.

In this case,

$$E = \sum_{1\leq i<j\leq n} (\mid x_i + l_i - x_j - l_j \mid -d_{ij})^2 + \lambda \sum_{i\in I^+} l_i - \lambda \sum_{i\in I^-} l_i\cdot$$

Let $\lambda \geq 0$, and $\gamma_1, \gamma_2 \dots \gamma_n$, n positive real numbers. The Lagrangian of the problem becomes:

$$L(l_1, \dots, l_n, \gamma_1, \dots, \gamma_n, \lambda) = \sum_{1\leq i<j\leq n} (\mid x_i + l_i - x_j - l_j \mid -d_{ij})^2 + \lambda \sum_{i\in I^+} l_i - \lambda \sum_{i\in I^-} l_i - \sum_{i\in I^+} \gamma_i l_i + \sum_{i\in I^-} \gamma_i l_i\cdot \tag{2.3}$$

By hypothesis of sets I^+ and I^- we have $\forall i = 1 \dots n$, $|l_i| > 0$ so that $\gamma_i = 0$. Since $x_i + l_i \leq x_j + l_j$, for all $1 \leq i < j \leq n$ from the hypothesis of the optimum we obtain the distance $\delta_{ij} = x_j + l_j - x_i - l_i$.

Let κ_1 and κ_2 two points such that $\kappa_1 \in I^+$ and $\kappa_2 \in I^-$. The first order optimality conditions give:

$$\begin{aligned}
\frac{\partial L}{\partial l_{\kappa_1}} &= 2\sum_{t=1}^{\kappa_1}(x_{\kappa_1} + l_{\kappa_1} - x_t - l_t - d_{t\kappa_1}) - 2\sum_{t=\kappa_1+1}^{n}(x_t + l_t - x_{\kappa_1} - l_{\kappa_1} - d_{t\kappa_1}) + \lambda = 0 \\
&= \sum_{t=1}^{n}(x_{\kappa_1} + l_{\kappa_1} - x_t - l_t) - \sum_{t=1}^{\kappa_1} d_{t\kappa_1} + \sum_{t=\kappa_1+1}^{n} d_{t\kappa_1} + \frac{\lambda}{2} = 0, \qquad (2.4) \\
\frac{\partial L}{\partial l_{\kappa_2}} &= 2\sum_{t=1}^{\kappa_2}(x_{\kappa_2} + l_{\kappa_2} - x_t - l_t - d_{t\kappa_2}) - 2\sum_{t=\kappa_2+1}^{n}(x_t + l_t - x_{\kappa_2} - l_{\kappa_2} - d_{t\kappa_2}) - \lambda = 0 \\
&= \sum_{t=1}^{n}(x_{\kappa_2} + l_{\kappa_2} - x_t - l_t) - \sum_{t=1}^{\kappa_2} d_{t\kappa_2} + \sum_{t=\kappa_2+1}^{n} d_{t\kappa_2} - \frac{\lambda}{2} = 0\cdot \qquad (2.5)
\end{aligned}$$

Subtracting equations (2.4) and (2.5) gives:

$$\begin{aligned}
0 &= \sum_{t=1}^{n}(x_{\kappa_1} + l_{\kappa_1} - x_{\kappa_2} - l_{\kappa_2}) + \sum_{t=1}^{\kappa_1}(d_{t\kappa_2} - d_{t\kappa_1}) + \sum_{t=\kappa_2}^{n}(d_{t\kappa_1} - d_{t\kappa_2}) + \lambda \\
&\leq \sum_{t=1}^{n}(x_{\kappa_1} + l_{\kappa_1} - x_{\kappa_2} - l_{\kappa_2}) + \sum_{t=1}^{\kappa_1} d_{\kappa_1\kappa_2} + \sum_{t=\kappa_2}^{n} d_{\kappa_1\kappa_2} + \lambda \text{ by triangle inequality} \\
&\leq n(x_{\kappa_1} + l_{\kappa_1} - x_{\kappa_2} - l_{\kappa_2}) + nd_{\kappa_1\kappa_2} + \lambda\cdot \qquad (2.6)
\end{aligned}$$

By hypothesis, for all $i = 1, \dots, n$ we have $|l_i| \leq \epsilon$ so:

$$\begin{aligned}
\lambda &\geq n(x_{\kappa_2} + l_{\kappa_2} - x_{\kappa_1} - l_{\kappa_1}) - nd_{\kappa_1\kappa_2} \\
\Rightarrow \lambda &\geq n(x_{\kappa_2} - x_{\kappa_1} - 2\epsilon) - nd_{\kappa_1\kappa_2}
\end{aligned}$$

So, as $x_{\kappa_1} < x_{\kappa_2}$ we obtain:

$$\lambda \geq n(d^0_{\kappa_1\kappa_2} - d_{\kappa_1\kappa_2} - 2\epsilon)$$

which concludes the proof.

In Lemma 2.3, we have proved, in dimension p, the same context of Lemma 2.2 that in a specific case, λ must not be too large.

Lemma 2.3. *In dimension p, for $\varepsilon > 0$, if $\lambda < \frac{n-1}{2}\varepsilon$ and under conditions, for all $1 \leq i < j \leq n$ and $k = 1, \ldots, n$:*

- $|\|X_i - X_j\|_2 - d_{ij}| > \varepsilon$,
- $\exists\ L_1, \ldots, L_n$ *such as* $\|L_k\|_2 < \varepsilon$,
- $\|X_i + L_i - X_j - L_j\|_2 = d_{ij}$,

then a solution of problem (P) can be found such that $E(L) < E(0)$.

Proof. By hypothesis we have considered that for $1 \leq i < j \leq n$, $\|X_i + L_i - X_j - L_j\|_2 - d_{ij} = 0$. Using the fact that $|\|X_i - X_j\|_2 - d_{ij}| > \varepsilon$, we obtain:

$$\sum_{1\leq i<j\leq n} (\|X_i - X_j\|_2 - d_{ij})^2 > \frac{n-1}{2} n\varepsilon^2. \tag{2.7}$$

From the assumption of displacements L_k, we obtain:

$$\sum_{1\leq i<j\leq n} \|X_i + L_i - X_j - L_j\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \|L_i\|_2 < n\lambda\varepsilon.$$

As $\lambda < \frac{n-1}{2}\varepsilon$, this leads to

$$\sum_{1\leq i<j\leq n} \|X_i + L_i - X_j - L_j\|_2 - d_{ij})^2 + \lambda \sum_{i=1}^{n} \|L_i\|_2 < \sum_{i<j} (\|X_i - X_j\|_2 - d_{ij})^2.$$

Hence $E(L) < E(0)$ which concludes the proof. □

Thus, from these two lemmas we can see that the order of magnitude of λ should be around n times ϵ, which is intuitively compared to the amplitude of random movements.

2.3 Application

In this section, penalized MDF has been applied to three different proteins to detect important movements in their tridimensional structure. Each time, the same protein is presented in two different conformations; for example, the three dimensional structure of a nuclear receptor before and after ligand binding. Data used in this study were collected

from Protein Data Bank (PDB) entries. These files contain the Cartesian coordinates obtained by X-ray crystallography for each atom in the protein. The amino acid positions are given by the coordinates of alpha carbons (C_α) which constitute the protein main chain.

Penalized MDF needs a reference matrix and a coordinate matrix. For two different structures of the same protein, the coordinate matrix is given by the C_α coordinates of one structure and the reference matrix is given by the Euclidean distances between the C_α of the second structure. The distance between two amino acids is given by: $d_{ij} = \sqrt{\sum_{k=1}^{p}(x_{ik} - x_{jk})^2}$, $\forall i, j \in 1 \dots n$ and $p = 3$. After matrices construction, we apply penalized MDF. The optimization problem (P) is the following:

$$(P) : \begin{cases} \min\limits_{L_1,\dots,L_n \in \mathbb{R}^p} & \sum\limits_{1 \le i < j \le n} (\|X_i + L_i - X_j - L_j\|_2 - d_{ij})^2 + \lambda\gamma \sum\limits_{i=1}^{n} \|L_i\|_2 + \lambda(1-\gamma) \sum\limits_{i=1}^{n} \|L_i\|_0 \\ s.t & l_{ik} \in [-(x_{k_{max}} - x_{k_{min}}); x_{k_{max}} - x_{k_{min}}] \end{cases}$$

where, $\forall k = 1 \dots p$, $x_{k_{max}} = \max\limits_i \{x_{ik}\}$ and $x_{k_{min}} = \min\limits_i \{x_{ik}\}$.

This problem is a non-linear optimization problem and it is clear that is non convex problem. The Nlopt library [13] has been used to solve it. Numerous algorithms exist for solving such non-linear optimization problems. We choose the best of them in order to minimize the objective function. A global optimization then a local optimization are applied to search the best solution.
For global optimization, we choose DIRECT-L algorithm (DIviding RECTangles) [6] which is a modified version of a partitioning algorithm called DIRECT algorithm [14] that samples points in a given domain and then refines the search domain at each iteration based on the function values at the sampled points . For local optimization, we choose SBPLX algorithm, this algorithm is based on SUBPLEX method [25] that is a generalization of NELDER-MEAD SIMPLEX. The global solution is considered as initial solution for local optimization.

For the choice of parameter λ, recall that by lemmas 2.2 and 2.3 we have seen that the order of magnitude of λ is n times the amplitude of random movements, that we take equal to 0.5Å for the present application. Concerning parameter γ, the following values have been used (0, 0.1, 0.3, 0.5, 0.8, 1) focusing then on 0.5 which gave the best results.

To have a threshold to determine if a movement is important, we compare the displacement norm computed after penalized MDF given by: $\|L_i\|_2$ with the standard deviation σ_i for each point i. For this, we use the known B-factor (or temperature factor) of each atom i which indicates the true static or dynamic mobility of an atom [24] given by: $B_i = 8\pi^2 d_{mi}^2$, to infer the mean displacement d_{mi} of atom i, $\forall i = 1, \dots, n$.
Besides, $d_{mi}^2 = E(\|X_i - \mu_i\|^2)$ with $X_i \rightsquigarrow \mathcal{N}(\mu_i; \sigma_i^2 I_3)$ and μ_i the mean coordinates for each atom i. Then, $d_{mi}^2 = \sigma_i^2 E(\frac{\|X_i - \mu_i\|^2}{\sigma_i^2}) = 3\sigma_i^2$ as $\frac{\|X_i - \mu_i\|^2}{\sigma_i^2} \rightsquigarrow \chi_3^2$. Thus, $\sigma_i = \frac{d_{mi}}{\sqrt{3}}$.

In our application, we suppose that the value $2\sigma_i$ is high enough to detect important displacements.

Penalized MDF has been applied to three proteins: human estrogen nuclear receptor (ER), Ferrichrome-iron receptor (FhuA), and aspartyl-tRNA synthetase (AspRS). For each protein, we compare, in Section 2.4, our results with those obtained by Procrustes analysis and DynDom.

2.3.1 Human Estrogen Receptor Protein

Human Estrogen Receptor (ER) is a nuclear estrogen receptor composed of several functional domains that serve specific roles. The ligand binding domain (LBD) region corresponds to the ligand binding domain, it is composed of 12 helices and is responsible of most of the functions activated by ligand binding [15]. Many experiments demonstrate that their C-terminal Helix (H12) is more flexible without ligand, this increased mobility being correlated with transcription repression and human diseases [2].
Penalized MDF has been used to compare the conformation with and without ligand.
The PDB entry 3ERT gives the Cartesian coordinates of amino acids of this receptor after ligand binding and 1X7R gives the coordinates before ligand binding. The reference matrix is computed from the coordinates of 1X7R.

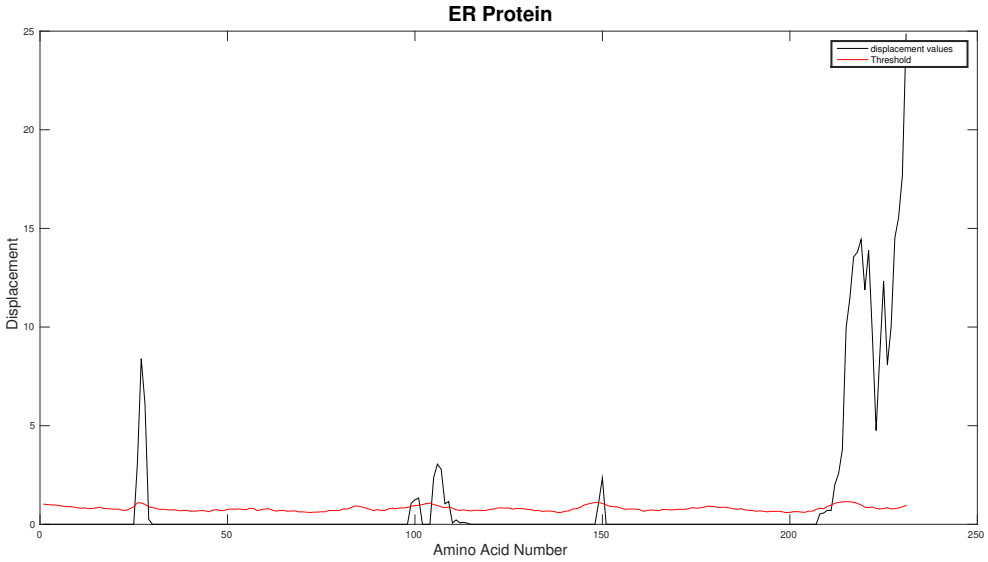


Figure 2.4: Displacement of each amino acid for Human estrogen receptor with $\lambda = \frac{n}{2}$ and $\gamma = 0.5$. The x-axis indicates the amino acids, and the y-axis indicates the displacement values. The amino acids number 26, 27, 28 and $214 - 231$ are are known to undergo important displacements.

The dashed plot in figure 2.4 shows the $2\sigma_i$ threshold for each point. Important displacements are detected at amino acids number $26, 27, 28$, and at the end of the sequence

formed by the amino acids number 214 to 231. Concerning the others detected amino acids, the value of the difference between the displacement after penalized MDF and $2\sigma_i$ is smaller than the other detected amino acids, so we can neglect these small detected displacements.

The sequence 214 to 231 corresponds to the helix H12. This result is confirmed by Mueller-Fahrnow and Egner [20] who note that the position of this helix depends on the presence or absence of a ligand. Concerning positions $26, 27$ and 28, they correspond to the sequence "SEA" which is apical of helix H3. After ligand binding, the helix H12 lies over the ligand-binding cavity and forms, together with H3 and H5, a binding surface for transcriptional co-regulators [27], which explains the movements of the helix H3.

2.3.2 Ferrichrome-iron Receptor Protein

FhuA is an outer membrane receptor protein of *Escherichia-coli* bacteries. FhuA and other proteins like FecA and FepA are ferrichrome transporters [5]. It acts as an energy dependent protein channel whose primary function is to transport ferrichrome across the outer membrane of E.coli.

X-ray analysis at 2.7Å resolution reveals two distinct conformations in the presence and absence of ferrichrome [16]. Penalized MDF has been applied to compare the two conformations, the coordinate matrix has been constructed from the PDB entry 1by3 of FhuA and the reference matrix has been calculated on the coordinates of C_α from the PDB entry 1by5 of FhuA+F. Figure 2.5 depicts important displacements between 4 and 18Å for the 10 first amino acids of the amino acid sequence of FhuA protein.

This result is confirmed by biology [16, 5] which shows that N-terminus has moved after ligand binding. Indeed, Molecular dynamics simulation method has been applied to detect the movements in FhuA protein [5]. Upon binding of ferrichrome, the formation of ligand-protein contacts induces a subtle conformational change in the segments S3 and S4 that yields an important displacement of 17Å for the segment S1 in the N-terminal. Additionally, [16] indicates that the comparison of the liganded with the unliganded state shows that the location of the N-terminus shifts considerably after ferrichrome binding.

2.3.3 Aspartyl-tRNA Synthetase Protein

Aspartyl-tRNA synthetase (AspRS) is a member of the aminoacyl-tRNA synthetases family, an ubiquitous enzyme family which translates the genetic code by linking a given amino acid to its cognate tRNA molecule. The AspRS is made of two monomers and binds two ATP, two aspartic acids and two tRNA molecules. The amino acid attachment reaction is performed only when all substrates are correctly recognized and bound to the two monomers.

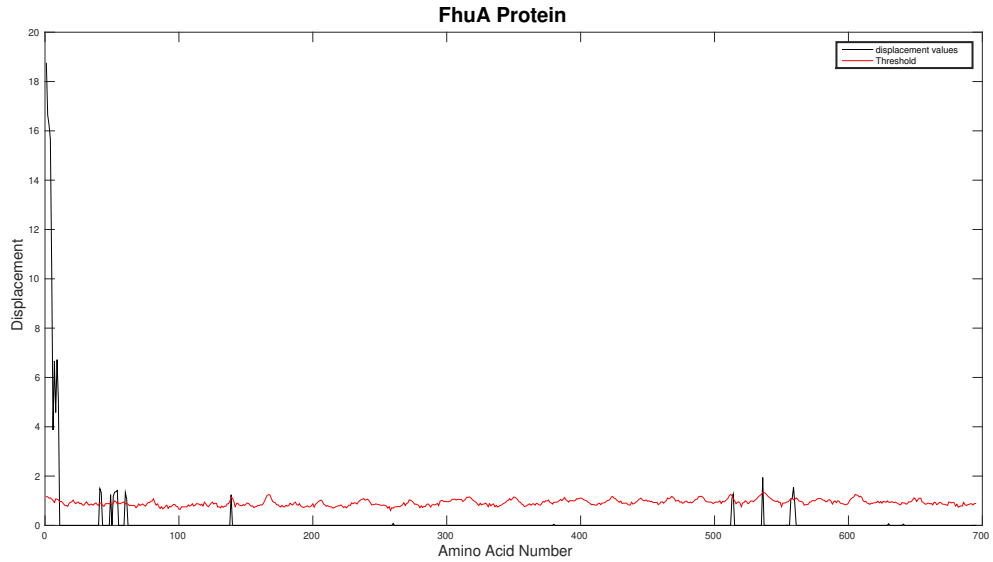


Figure 2.5: Displacements of the amino acids for FhuA protein. Important displacements are located in the N-terminus (the first amino acids).

Here, we have considered the complexed AspRS (PDB entry 1IL2) made of two monomers of the E. coli AspRS protein, two $tRNA^{Asp}$ from yeast, and AMPPcP, a substrate analogue. This complex is biologically non-productive as the bound $tRNA^{Asp}$ is from yeast origin and thus badly recognized by the E. coli enzyme. However, one of the two tRNAs is correctly bound to the active site (chains A/C) whereas the second lies in the solvent (chain B/D). This complex constitutes a good candidate to study conformational changes at the enzyme level upon tRNA binding by comparing the two monomers of the protein.

Penalized MDF method has been applied to compare chains A and B; the reference matrix has been computed from the C_α coordinates of chain B and the target matrix computed from the C_α coordinates of chain A. Results are presented in figure 2.6 where it appears that important displacements occur in positions 220, $330-332$, $335-337$ $342-343$ and 468. Penalized MDF finds moving positions inside the GAD domain of the monomer, this domain has an unknown function, but moves towards the active site upon binding of the acceptor stem of a tRNA molecule so as to pinch it through a network of water molecules [19].

2.4 Comparaison with other methods

In this section, we have compared penalized MDF results with Procrustes analysis and DynDom results.

Firstly, we have compared our results with Procrustes analysis results. Figure 2.7 depicts

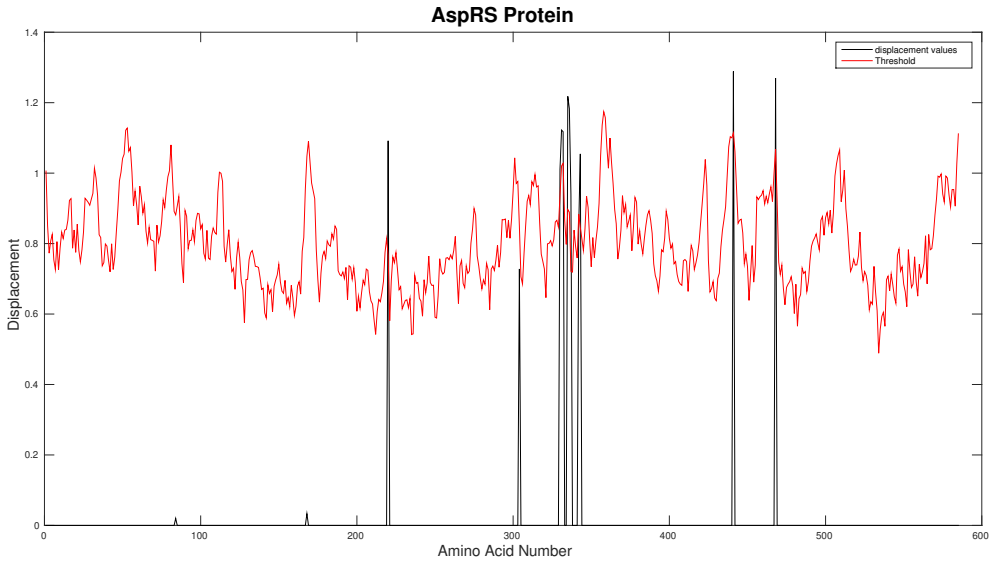


Figure 2.6: Displacements of the amino acids for AspRS protein. Amino acids having distances larger than the threshold are: 220, $330-332$, $335-337$, $342-343$, 441 and 468.

the movements of amino acids of the three proteins studied above using two methods. Procrustes analysis detects more displacements than penalized MDF method and a number of them are detected as important displacements mainly for ER and AspRS proteins. Secondly, we have used the web application DynDom available over the internet (`http://fizz.cmp.uea.ac.uk/dyndom/dyndomMain.do`) to obtain DynDom results. No domains movements are detected by using DynDom for ER and FhuA proteins, whereas using penalized MDF, we have succeed in finding the residues that move according to the known biological literature. For AspSR protein, DynDom finds the following positions $287-308$, $312-315$, 321, $323-404$ and $412-425$. Both DynDOM and penalized MDF find moving positions inside the GAD domain of the monomer. Interestingly, penalized MDF finds also position 220, a conserved position in bacterial AspRS that controls the recognition of the acceptor stem of the tRNA. Position 468 is displaced because of crystal packing contacts; this has no biological significance.

2.5 Reference matrix reduction

In order to simplify the optimization problem and reduce the computing time, it is necessary to reduce the reference matrix by conserving as much as possible the information given by this matrix. For this task, we consider our reference matrix as a graph with $\frac{n(n-1)}{2}$ edges, each point is a vertex and each distance is an edge. Our objective is to reduce this graph to another one with fewer edges.

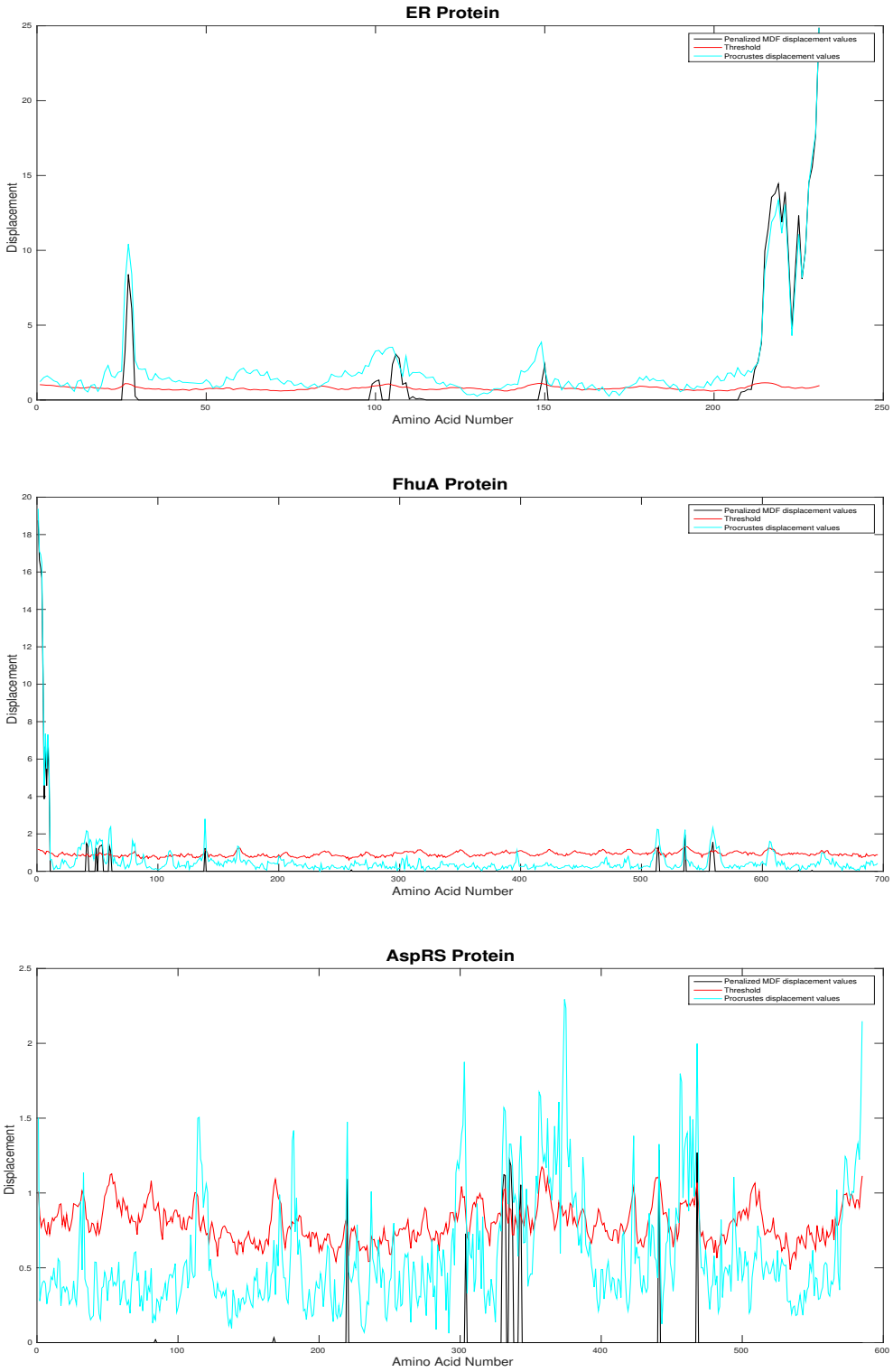


Figure 2.7: The displacements of amino acids after Procrustes analysis are given by the cyan line plot and those after penalized MDF are given by the black line plot. The threshold is the red line plot. We notice that using Procrustes analysis we detect more amino acids moved which are in reality not moved.

The idea is to divide the n points into q clusters using k-means. The squared distance between i and j is given by: $d_{ij}^2 = \|Y_i - Y_j\|_2^2$ where $Y_1, \ldots, Y_n$ are the coordinates of the n points obtained from the reference matrix.

The complete graph is replaced by a reduced graph with $\frac{(q-1)}{2} + (n-q)$ edges, where the distances between cluster centers and between cluster centers and their points are kept. Thus, the complete reference matrix with $\frac{n(n-1)}{2}$ is replaced by another one with $\frac{q(q-1)}{2} + (n-q)$ distances.

Using k-means method, we partition the n points into q clusters and then the points are supposed to be uniformly distributed across the clusters. The center of each cluster is given by the nearest neighbour of the center of gravity of clusters. Each cluster is considered as an hypersphere and using the measure concentration on the sphere, for dimension d large, we can consider that most of the points are concentrated in the equator of the sphere S^{d-1} in $\mathbb{R}^d$ [18]. Figure 2.8 shows the width of the band around the equator that contains 90% of the measure for different dimensions. If the width of the gray stripe is 2ω then:

$$P[\{x \in S^{d-1} : -\omega \leq x_n \leq \omega\}] = 0.9\cdot$$

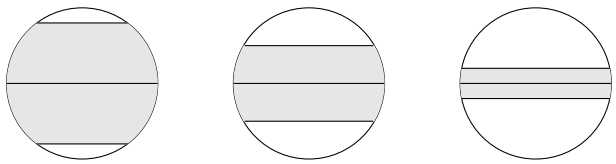


Figure 2.8: Strip around the equator containing 90% of the area.

Using this result, we can consider that each edge is orthogonal to the edge that connects two cluster centers. Figure 2.9 presents the distance between two points which are located in two different clusters.

So we can approximate the squared distance d_{ij}^2 between i and j by:

$$\|Y_i - Y_{O_k}\|_2^2 + \|Y_{O_k} - Y_{O_l}\|_2^2 + \|Y_j - Y_{O_l}\|_2^2,$$

where O_k and O_l are respectively the centers of clusters C_k and C_l.

Figure 2.9: The black line is the distance between two points i and j which are located in two different clusters. The gray band is the concentration measure band.

The idea using k-means is to keep the minimal number of distances between points that is equivalent to minimize the distances between nodes in the reduced graph. Therefore the problem is to minimize the sum of the squared distances between all points. The objective function of this minimization problem is the following :

$$V = \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \sum_{k=1}^{q} \sum_{l=1}^{q} \left(\|Y_i - Y_{O_k}\|_2^2 z_{ik} + \|Y_{O_k} - Y_{O_l}\|_2^2 z_{ik} z_{jl} + \|Y_j - Y_{O_l}\|_2^2 z_{jl} \right),$$

with for $i = 1, \ldots, n$ and $k = 1, \ldots, q$, $z_{ik} = \begin{cases} 1 & \text{if } i \in C_k \\ 0 & \text{otherwise.} \end{cases}$

By developing V we obtain:

$$V = 2q(n-1) \sum_{i=1}^{n} \sum_{k=1}^{q} \|Y_i - Y_{O_k}\|_2^2 z_{ik} + \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \sum_{k=1}^{q} \sum_{l=1}^{q} \|Y_{O_k} - Y_{O_l}\|_2^2 z_{ik} z_{jl}.$$

Besides, by noting O as the center of gravity of the clusters centers:

$$\begin{aligned}
&\sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \sum_{k=1}^{q} \sum_{l=1}^{q} \|Y_{O_k} - Y_{O_l}\|_2^2 z_{ik} z_{jl} \\
\approx &\sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \sum_{k=1}^{q} \sum_{l=1}^{q} \left(\|Y_{O_k} - Y_O\|_2^2 + \|Y_O - Y_{O_l}\|_2^2 z_{ik} z_{jl} \right) \quad \text{provided that } OO_k \perp OO_l \\
\approx &q(n-1) \sum_{i=1}^{n} \sum_{k=1}^{q} \|Y_{O_k} - Y_O\|_2^2 z_{ik} + q(n-1) \sum_{j=1}^{n} \sum_{l=1}^{q} \|Y_O - Y_{O_l}\|_2^2 z_{jl} \\
\approx &2q(n-1) \sum_{i=1}^{n} \sum_{k=1}^{q} \|Y_{O_k} - Y_O\|_2^2 z_{ik}.
\end{aligned}$$

Consequently,

$$\begin{aligned}
V &\approx 2q(n-1) \sum_{i=1}^{n} \sum_{k=1}^{q} \|Y_i - Y_{O_k}\|_2^2 z_{ik} + 2q(n-1) \sum_{i=1}^{n} \sum_{k=1}^{q} \|Y_{O_k} - Y_O\|_2^2 z_{ik} \\
&\approx 2q(n-1) \sum_{i=1}^{n} \|Y_i - Y_O\|_2^2.
\end{aligned}$$

Thus, the points total variation V_T is proportional to $\sum_{i=1}^{n} \|Y_i - Y_O\|_2^2$, so that V is a constant and it is not necessary to minimize it. For the choice of the number of clusters, we take it at most equal to $\sqrt{n}$.

Penalized MDF with reduced matrix reference requires a novel penalization parameter calculation as less variables are involved.

2.5.1 Novel Penalization Parameter Calculation

In this case, the objective function of penalized MDF problem is the following:

$$E_r = \sum_{i=1}^{n}\sum_{k=1}^{q}\left(\|X_i + L_i - X_{O_k} - L_{O_k}\|_2 - d_{iO_k}\right)^2 z_{ik} + \lambda\sum_{i=1}^{n}\|L_i\|_2 + \sum_{k=1}^{q}\sum_{l=k+1}^{q}\left(\|X_{O_k} + L_{O_k} - X_{O_l} - L_{O_l}\|_2 - d_{O_kO_l}\right)^2. \tag{2.8}$$

The optimization problem here is:

$$(P_r) : \min_{L_1,\ldots,L_n\in\mathbb{R}^p} E_r.$$

Let give firstly some notations and assumptions. In one dimension, noted o_k and o_l respectively the centers of clusters C_k and C_l, we suppose that $x_{o_k} + l_{o_k} \leq x_{o_l} + l_{o_l}$, for $k = 1,\ldots,q$ and $l = k+1,\ldots,q$. Let:

- I^+ the set of all points i that move in the positive direction, we note $I^+ = \{i|l_i > 0\}$.
- I^- the set of all points i that move in the negative direction, we note $I^- = \{i|l_i < 0\}$.
- A^+ the set of all points i such that $x_i + l_i - x_{o_k} - l_{o_k} > 0$.
- A^- the set of all points i such that $x_i + l_i - x_{o_k} - l_{o_k} < 0$.
- $n_t^{UV} = \mathrm{Card}(U \cap V)_t$, the number of points in the set $U \cap V$ and cluster C_t.

Lemma 2.4. *In one dimension, at the optimum if for two successive cluster centers o_{κ_1} and o_{κ_2} ($\kappa_2 = \kappa_1 + 1$) having $x_{o_{\kappa_1}} < x_{o_{\kappa_2}}$ and moving such that $l_{o_{\kappa_1}} > 0$ and $l_{o_{\kappa_2}} < 0$; the parameter λ is such that $\lambda < \frac{2q}{N+2}(d^0_{o_{\kappa_1}o_{\kappa_1}} - d_{o_{\kappa_1}o_{\kappa_2}} - 2\epsilon)$ then $\exists\, m$, with $m \in \{1,\ldots,n\}$, such that $|l_m| > \epsilon$, where $d^0_{o_{\kappa_1}o_{\kappa_2}}$ is the initial distance computed on the target matrix and $N = n_{\kappa_1}^{A^+I^+} - n_{\kappa_1}^{A^+I^-} + n_{\kappa_1}^{A^-I^+} - n_{\kappa_1}^{A^-I^-} - n_{\kappa_2}^{A^+I^+} + n_{\kappa_2}^{A^+I^-} - n_{\kappa_2}^{A^-I^+} + n_{\kappa_2}^{A^-I^-}$.*

Proof. We prove it by contrapositive assuming that for all i we have $|l_i| \leq \epsilon$. The Lagrangian of the optimization problem is the following:

$$\begin{aligned}
L(l_1,\ldots,l_n,\gamma_1,\ldots,\gamma_n,\lambda) = &\sum_{i\in A^+}\sum_{k=1}^{q}\left(x_i + l_i - x_{o_k} - l_{o_k} - d_{io_k}\right)^2 z_{ik} \\
&+ \sum_{i\in A^-}\sum_{k=1}^{q}\left(x_{o_k} + l_{o_k} - x_i - l_i - d_{io_k}\right)^2 z_{ik} \\
&+ \sum_{k=1}^{q}\sum_{l=k+1}^{q}\left(x_{o_l} + l_{o_l} - x_{o_k} - l_{o_k} - d_{o_ko_l}\right)^2 \\
&+ \lambda\sum_{i\in I^+} l_i - \lambda\sum_{i\in I^-} l_i - \sum_{i\in I^+}\gamma_i l_i + \sum_{i\in I^-}\gamma_i l_i
\end{aligned}$$

By hypothesis, for all $i = 1\ldots n$ we have $|l_i| > 0$ so that $\gamma_i = 0$. The first order optimality conditions give:

For $i \in C_k$ and $i \neq o_k$, $\dfrac{\partial L}{\partial l_i} = 0$ gives :

$$\text{if } i \in I^+ \cap A^+, \quad x_i + l_i - x_{o_k} - l_{o_k} - d_{io_k} = -\frac{\lambda}{2} \tag{2.9}$$

$$\text{if } i \in I^+ \cap A^-, \quad x_i + l_i - x_{o_k} - l_{o_k} + d_{io_k} = -\frac{\lambda}{2} \tag{2.10}$$

$$\text{if } i \in I^- \cap A^+, \quad x_i + l_i - x_{o_k} - l_{o_k} - d_{io_k} = \frac{\lambda}{2} \tag{2.11}$$

$$\text{if } i \in I^- \cap A^-, \quad x_i + l_i - x_{o_k} - l_{o_k} + d_{io_k} = \frac{\lambda}{2} \tag{2.12}$$

For the two centers o_{κ_1} and o_{κ_2} of clusters C_{κ_1} and C_{κ_2} such that $o_{\kappa_1} \in I^+$ and $o_{\kappa_1} \in I^-$ we have:

$$\begin{aligned}\frac{\partial L}{\partial l_{o_{\kappa_1}}} =& -\sum_{t\in A^+}(x_t + l_t - x_{o_{\kappa_1}} - l_{o_{\kappa_1}} - d_{to_{\kappa_1}})z_{t\kappa_1} + \sum_{t\in A^-}(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_t - l_t - d_{to_{\kappa_1}})z_{t\kappa_1} \\ &+ \sum_{u=1}^{q}(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_{o_u} - l_{o_u}) - \sum_{u=1}^{\kappa_1} d_{o_u o_{\kappa_1}} + \sum_{u=\kappa_1+1}^{q} d_{o_u o_{\kappa_1}} + \frac{\lambda}{2} = 0 \cdot \end{aligned} \tag{2.13}$$

$$\begin{aligned}\frac{\partial L}{\partial l_{o_{\kappa_2}}} =& -\sum_{t\in A^+}(x_t + l_t - x_{o_{\kappa_2}} - l_{o_{\kappa_2}} - d_{to_{\kappa_2}})z_{t\kappa_2} + \sum_{t\in A^-}(x_{o_{\kappa_2}} + l_{o_{\kappa_2}} - x_t - l_t - d_{to_{\kappa_2}})z_{t\kappa_2} \\ &+ \sum_{u=1}^{q}(x_{o_{\kappa_2}} + l_{o_{\kappa_2}} - x_{o_u} - l_{o_u}) - \sum_{u=1}^{\kappa_2} d_{o_u o_{\kappa_2}} + \sum_{u=\kappa_2+1}^{q} d_{o_u o_{\kappa_2}} - \frac{\lambda}{2} = 0 \cdot \end{aligned} \tag{2.14}$$

Substracting equations (2.13) and (2.14) and using equations (2.9), (2.10), (2.11) and (2.12), we obtain:

$$\begin{aligned}0 =& \left(n_{\kappa_1}^{A^+I^+} - n_{\kappa_1}^{A^+I^-} + n_{\kappa_1}^{A^-I^+} - n_{\kappa_1}^{A^-I^-} - n_{\kappa_2}^{A^+I^+} + n_{\kappa_2}^{A^+I^-} - n_{\kappa_2}^{A^-I^+} + n_{\kappa_2}^{A^-I^-}\right)\frac{\lambda}{2} \\ &+ \sum_{u=1}^{q}(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_{o_{\kappa_2}} - l_{o_{\kappa_2}}) - \sum_{u=1}^{\kappa_1} d_{o_u o_{\kappa_1}} + \sum_{u=\kappa_1+1}^{q} d_{o_u o_{\kappa_1}} + \sum_{u=1}^{\kappa_2} d_{o_u o_{\kappa_2}} - \sum_{u=\kappa_2+1}^{q} d_{o_u o_{\kappa_2}} + \lambda \\ \leq& (N+2)\frac{\lambda}{2} + \sum_{u=1}^{q}(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_{o_{\kappa_2}} - l_{o_{\kappa_2}}) - \sum_{u=1}^{\kappa_1} d_{o_u o_{\kappa_1}} + \sum_{u=\kappa_1+1}^{q} d_{o_u o_{\kappa_1}} + \sum_{u=1}^{\kappa_1} d_{o_u o_{\kappa_2}} \\ &+ \sum_{u=\kappa_1+1}^{\kappa_2} d_{o_u o_{\kappa_2}} - \sum_{u=\kappa_2+1}^{q} d_{o_u o_{\kappa_2}} \text{ as } \kappa_1 < \kappa_2 \\ \leq& (N+2)\frac{\lambda}{2} + \sum_{u=1}^{q}(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_{o_{\kappa_2}} - l_{o_{\kappa_2}}) - \sum_{u=1}^{\kappa_1} d_{o_u o_{\kappa_1}} + \sum_{u=\kappa_1+1}^{q} d_{o_u o_{\kappa_2}} + \sum_{u=\kappa_1+1}^{q} d_{o_{\kappa_1} o_{\kappa_2}} \\ &+ \sum_{u=1}^{\kappa_1} d_{o_u o_{\kappa_1}} + \sum_{u=1}^{\kappa_1} d_{o_{\kappa_1} o_{\kappa_2}} + \sum_{u=\kappa_1+1}^{\kappa_2} d_{o_u o_{\kappa_2}} - \sum_{u=\kappa_2+1}^{q} d_{o_u o_{\kappa_2}} \text{ by triangle inequality} \end{aligned}$$

$$\leq (N+2)\frac{\lambda}{2} + \sum_{u=1}^{q}(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_{o_{\kappa_2}} - l_{o_{\kappa_2}}) + \sum_{u=1}^{q} d_{o_{\kappa_1}o_{\kappa_2}} \text{ as } \kappa_1 \text{ and } \kappa_2 \text{ are two successive points}$$

$$\leq (N+2)\frac{\lambda}{2} + q(x_{o_{\kappa_1}} + l_{o_{\kappa_1}} - x_{o_{\kappa_2}} - l_{o_{\kappa_2}}) + q d_{o_{\kappa_1}o_{\kappa_2}}$$

$$\leq (N+2)\frac{\lambda}{2} + q(x_{o_{\kappa_1}} - x_{o_{\kappa_2}} + 2\epsilon) + q d_{o_{\kappa_1}o_{\kappa_2}} \qquad \text{as } |l_i| \leq \epsilon$$

$$\leq (N+2)\frac{\lambda}{2} + q(d_{o_{\kappa_1}o_{\kappa_2}} - d^0_{o_{\kappa_1}o_{\kappa_2}} + 2\epsilon) \qquad \text{as } x_{o_{\kappa_1}} < x_{o_{\kappa_2}}.$$

Then

$$\lambda \geq \frac{2q}{N+2}\left(d^0_{o_{\kappa_1}o_{\kappa_2}} - d_{o_{\kappa_1}o_{\kappa_2}} - 2\epsilon\right),$$

which concludes the proof. □

2.5.2 Application

Penalized MDF method has been applied using reduced reference matrix. The optimization results were obtained much more quickly (even thirty times faster than using complete matrix for ER protein) and the amino acids movements detected are similar to those obtained with the complete reference matrix.

Penalized MDF method has been applied using reduced reference matrix. The optimization results were obtained much more quickly (even thirty times faster than using complete matrix for ER protein) and the amino acids movements detected are similar to those obtained with the complete reference matrix.

For the choice of λ, as N can be equal to 0 we take the smallest admissible value considering the examples at hand, that is 4. Thus we obtain $\lambda \leq \frac{q}{3}\left(d^0_{o_{\kappa_1}o_{\kappa_1}} - d_{o_{\kappa_1}o_{\kappa_1}} - 2\epsilon\right)$. Note that by increasing the value of N we decrease λ and more amino acids are moved and these movements can be also detected as important displacements. Concerning the amplitude of random movements, we take it equal to 0.5Å . The application results are given by figure 2.10.

The amino acids detected with the reduced reference matrix are approximately similar to those detected with the complete one with the exception of AspRS for which q should be taken higher.

2.6 Conclusion

The purpose of penalized MDF is to modify only the coordinates of amino acids that have significantly moved and fix the others. Penalization term and penalization parameters are crucial in the process of obtaining good results. This involves the choice of a penalty coefficient λ which is related to the minimum displacement.

Penalized MDF has been applied to three different proteins in order to find the residues that were affected by the interaction with other molecules. Comparison of our results

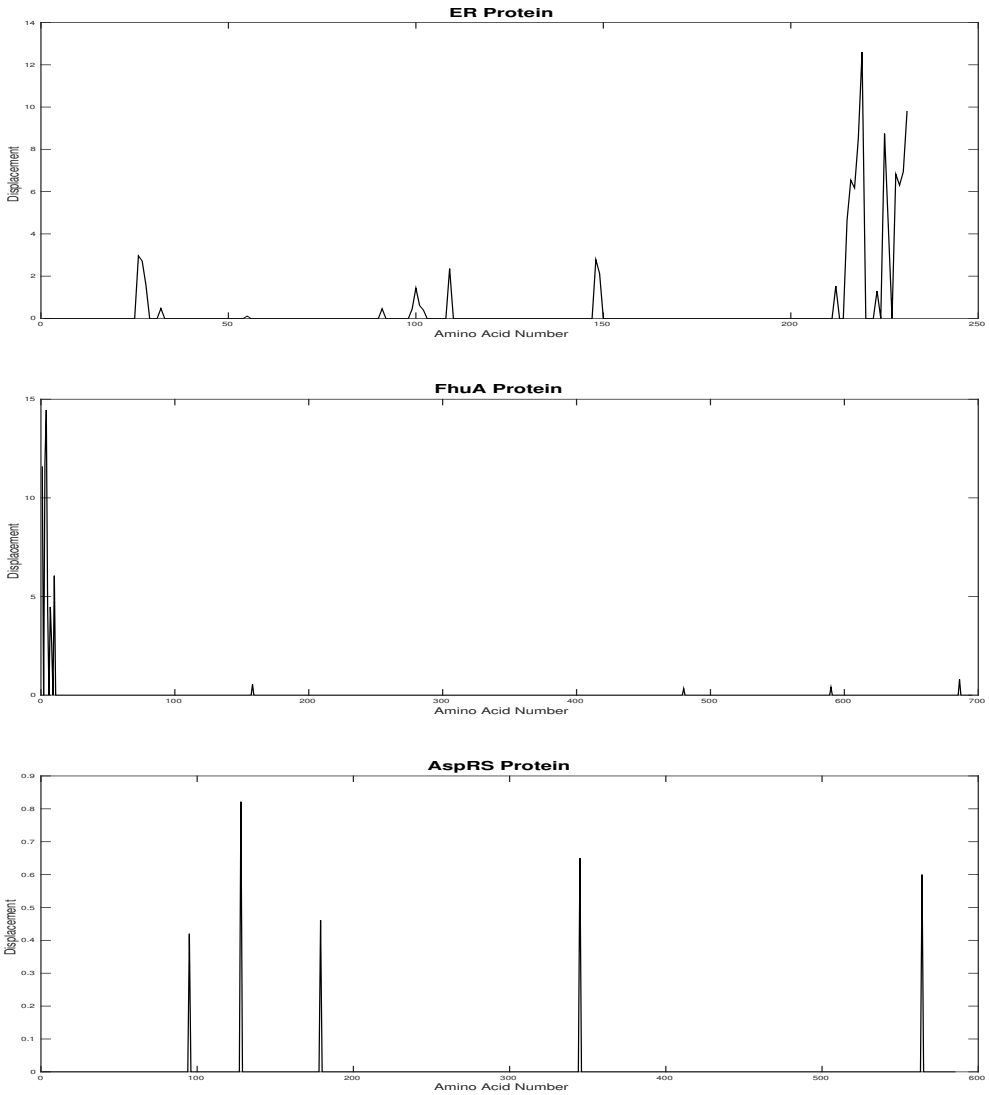


Figure 2.10: The displacements of amino acids for the three proteins studied above. The line plot of movements for the two proteins ER and FhuA are similar to that obtained in figures 2.4 and 2.5.

with Procrustes analysis, DynDom results and the literature reveals that penalized MDF is an efficient method for protein movement detection and fills a gap in the computational toolbox of molecular biologists.

Bibliography

[1] Adcock, S.A. and McCammon, J.A., Molecular Dynamics: Survey of Methods for Simulating the Activity of Proteins, Chemical Reviews, 106(2006), pp. 1589–1615.

[2] Batista, M.R.B. and Martinez, L., Dynamics of Nuclear Receptor Helix-12 Switch of Transcription Activation by Modeling Time-Resolved Fluorescence Anisotropy Decays, Biophysical journal, 105(2013), pp. 1670–1680.

[3] Berge, C., Froloff, N., Kalathur, RK., Maumy, M., Poch, O., Raffelsberger, W. and Wicker, N., Multidimensional fitting for multivariate data analysis, Journal of Computational Biology, 17(2010), pp. 723–732.

[4] De Groot, B. L., Hayward, S., van Aalten, D., Amadei, A. and Berendsen, H.J.C. , Domain motions in bacteriophage T4 lysozyme: a comparison between molecular dynamics and crystallographic data Proteins, 31(1998), pp. 116–127.

[5] Faraldo-Gomez J.D., Smith G.R. and Sansom M.S., Molecular dynamics simulations of the bacterial outer membrane protein FhuA: a comparative study of the ferrichrome-free and bound states, Biophysical journal, 85(2003), pp. 1406-1420.

[6] Gablonsky, J. M. and Kelley, C. T., A locally-biased form of the DIRECT algorithm, Global Optimization, 21(2001), pp. 27–37.

[7] Gower, J.C. and Dijksterhuis, G B., Procrustes Problems (2004), Oxford Statistical Science Series 30 .

[8] Gower, J.C., Statistical methods of comparing different multivariate analyses of the same data, Mathematics in the Archaeological and Historical Sciences. Edinburgh University Press, Edinburgh(1971) pp. 138–149.

[9] Green, B.F., The orthogonal approximation of an oblique structure in factor analysis, Psychometrika, 17(1952), pp. 429–440.

[10] Hammes-Schiffer, S. and Benkovic, SJ., Relating protein motion to catalysis, Annual Review of Biochemistry, 75(2006), pp. 519–541.

[11] Hayward, S., Kitao, A. and Berendsen, H.J.C., Model-free methods of analyzing domain motions in proteins from simulations: a comparison of normal mode analysis and molecular dynamics simulation, Proteins, 27(1997), pp. 425–437.

[12] Hayward, S. and Berendsen, H.J.C., Systematic analysis of domain motions in proteins from conformational change: new results on citrate synthase and T4 lysozyme, Proteins, 30(1998), pp. 144–154.

[13] Johnson, S.G., The NLopt nonlinear-optimization package (2008), available at http://ab-initio.mit.edu/nlopt

[14] Jones, D.R., Perttunen, C.D. and Stuckmann, B.E,, Lipschitzian optimization without the lipschitz constant,Optimization Theory and Applications, 79(1993), pp. 157-181.

[15] Kumar, R., Zakharov, M.N., Khan, S.H., Miki, R., Jang, H., Toraldo, G., Singh, R., Bhasin, S. and Jasuja, R., The Dynamic Structure of the Estrogen Receptor (2011), Journal of Amino Acids [online], DOI: 10.4061/2011/812540. Available at http://www.ncbi.nlm.nih.gov/pubmed/22312471

[16] Locher, K.P, Rees, B, Koebnik, R, Mitschler, A, Moulinier, L, Rosenbusch, J.P and Moras, D., Transmembrane Signaling across the Ligand-Gated FhuA Receptor: Crystal Structures of Free and Ferrichrome-Bound States Reveal Allosteric Changes, Cell press, 95(1998), pp. 771–778.

[17] Mardia, K.V., Kent, J.T. and Bibby, J.M., Multivariate Analysis (1979), Academic Press, New York.

[18] Matousek, J., Lectures on Discrete Geometry (2002), Springer, 212.

[19] Moulinier, L., Eiler, S., Eriani, G., Gangloff, J., Thierry, JC., Gabriel, K., McClain, WH. and Moras, D., The structure of an AspRS-tRNA(Asp) complex reveals a tRNA-dependent control mechanism, The EMBO Journal, 20(2001), pp. 5290–5301.

[20] Mueller-Fahrnow, A. and Egner, U., Ligand-binding domain of estrogen receptors, Current Opinion in Biotechnology, 10(1999), pp. 550–556.

[21] Nichols, W.L., Rose, G.D., Ten Eyck, L.F. and Zimm, B.H, Rigid domains in proteins: an algorithmic approach to their identification, Proteins, 23(1995), pp. 38–48.

[22] Petsko, G.A. and Ringe, D., Protein Structure and Function (2004), New science press.

[23] Poornam, GP., Matsumoto, A., Ishida, H. and Hayward, S., A method for the analysis of domain movements in large biomolecular complexes, Proteins, 76(2009), pp. 201–212.

[24] Rhodes, G., Judging the Quality of Macromolecular Models: A Glossary of Terms from Crystallography NMR and Homology Modeling (2000), available at http://spdbv.vital-it.ch/TheMolecularLevel/ModQual/.

[25] Rowan, T., Functional Stability Analysis of Numerical Algorithms, published Ph.D. diss., University of Texas at Austin (1990).

[26] Skjaervena, L., Hollupc, S.M. and Reutera, N., Normal mode analysis for proteins, Journal of Molecular Structure: Theochem, 898(2009), pp. 42–48.

[27] Sumbayev, V.V, Bonefeld-Jorgensen, E.C., Wind, T. and Andreasen, P.A., A novel pesticide-induced conformational state of the oestrogen receptor ligand-binding domain, detected by conformation-specific peptide binding, FEBS Letters, 579(2005), pp. 541–548.

[28] Teilum, K, Olsen, J.G and Kragelund, B.B, Functional aspects of protein flexibility, Cellular and Molecular Life Sciences, 66(2009), pp. 2231–2247.

[29] Wriggers, W. and Schulten, K., Protein domain movements: detection of rigid domains and visualization of hinges in comparisons of atomic coordinates, Proteins, 29(1997), pp. 1–14.

[30] Zou, H. and Hastie, T., Regularization and Variable Selection via the Elastic Net, Journal of the Royal Statistical Society, Series B, 67(2005), pp. 301–320.

Chapter 3

Random model for Multidimensional Fitting method

In chapter 2, multidimensional fitting method with a penalized version was shown. The displacement vectors are taken as deterministic vectors and the random effects that can be produced during the modification and can affect the interpretation of the modifications significance are not taken into account. In this chapter, we want to introduce the random effect in the model of MDF method to find then the real displacement vectors. The minimization of the mean square error between the new distances and the reference distances performed in the deterministic model of MDF, to obtain the optimal values of displacement vectors, cannot be applied for the random model as the objective function here is a random variable. Therefore, we want to use different ways to find these vectors. First of all, the random model of MDF is presented in Section 3.1, then in Sections 3.2 and 3.3, two ways to obtain the optimal values of displacement vectors are illustrated. After that, an application in the sensometrics domain is presented in Section 3.7 in order to fit the sensory profiles of products to consumers preferences of these products. Finally, we conclude our work in Section 3.8.

3.1 The random model of Multidimensional Fitting

In chapter 2 it was shown that the modification function f is written as follows: $f(X_i) = X_i + L_i$, where for $i = 1, \ldots, n$, the vectors X_i and L_i in $\mathbb{R}^p$ are, respectively, the coordinate and the displacement vectors of point i. The optimization problem of MDF is given by:

$$\Delta = \sum_{1 \leq i < j \leq n} (d_{ij} - a\ \delta_{ij})^2$$

with $\delta_{ij} = ||X_i + L_i - X_j - L_j||_2$, $a \in \mathbb{R}$ a scaling variable.

Owing to the negligence of random effects that can occur during modification, the interpretation of the displacements can be erroneous. Thereby, to tackle this problem we introduce the random effects in the modification function. So, the new modification func-

tion is given by:

$$f(X_i) = X_i + \theta_i + \varepsilon_i,$$

where θ_i and ε_i in $\mathbb{R}^p$ are, respectively, the fixed and random part of modification.
Contrary to what has been seen above, δ_{ij} here is a random variable and not a deterministic value, for all $1 \leq i < j \leq n$, so the error Δ cannot be minimized directly. Deterministic and stochastic optimization are presented to find the optimum value of vectors $(\theta_1, \ldots, \theta_n)$:

1- Deterministic optimization: by minimizing a function depending on vectors $(\theta_1, \ldots, \theta_n)$ with consideration that the components of vectors ε_i, for all $i = 1, \ldots, n$, are independently and identically normally distributed.

2- Stochastic optimization: by simulating the error Δ where the components of vectors ε_i for all $i = 1, \ldots, n$ are dependent and not normally distributed.

3.2 Calculation of $(\theta_1^*, \ldots, \theta_n^*)$ by minimization

In this section, we suppose that the components of vector ε_i denoted ε_{ik}, for all $i = 1, \ldots, n$ and $k = 1, \ldots, p$, are p-dimensional independent and identically distributed random variables where the vector ε_i is multivariate normally distributed with mean $\mathbb{E}(\varepsilon_i) = \mathbf{0}$ (the vector $\mathbf{0}$ in $\mathbb{R}^p$ is the null vector) and variance $\mathbb{V}ar(\varepsilon_i) = \sigma^2 I_p$ (σ is a strictly positive value to be fixed and I_p is the identity matrix).
We note Γ a $n \times n$ matrix that contains the distances between the points after modification. The objective function of the minimization problem called $g(D, \Gamma)$ can be expressed in different forms. We cite below some of them:

$$g: \ \mathcal{M}_{n\times n}(\mathbb{R}) \times \mathcal{M}_{n\times n}(\mathbb{R}) \longmapsto \mathbb{R}$$

$$(D, \Gamma) \longmapsto \begin{cases} \mathbb{E}(\|D - a\Gamma\|_2^2) \\ \text{med}(\|D - a\Gamma\|_2^2) \\ \text{minmax}(\|D - a\Gamma\|_2^2) \\ \|D - a\ \mathbb{E}(\Gamma)\|_2^2 \\ \|D - a\ \text{med}(\Gamma)\|_2^2 \\ \|\text{minmax}(D - a\Gamma)\|_2^2. \end{cases}$$

In our work, we are interested to take $g(D, \Gamma) = \mathbb{E}(\|D - a\Gamma\|_2^2)$. The expression of $\|D - a\Gamma\|_2^2$ noted Δ (as the mean square error cited above) can be rewritten as:

$$\Delta = \sum_{1\leq i<j\leq n} (d_{ij} - a\|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|_2)^2.$$

The problem here is to find the vectors $(\theta_1^*, \ldots, \theta_n^*)$ such that the minimum of $\mathbb{E}(\Delta)$ under $(\theta_1, \ldots, \theta_n)$ is reached. The initial optimization problem (P_0) is defined by:

$$(\mathrm{P}_0) : \min_{\theta_1,\ldots,\theta_n \in \mathbb{R}^p} \mathbb{E}(\Delta)\cdot$$

The optimal solution obtained from (P_0) is a solution assigns the smallest value to $\mathbb{E}(\Delta)$ but moves too many points. So, it is a good solution from minimization standpoint, but awkward from parsimony standpoint.

A new optimization problem is presented to find the optimal vectors $(\theta_1^*, \ldots, \theta_n^*)$ by taking into account the minimization of the expectation of Δ and the parsimonious choice of displacements. A natural approach to obtain such sparsity solution is to use the number of non-zero displacements as a penalty. So, a penalty term can be defined, using ℓ_0-norm, as $\sum_{i=1}^{n} \|\theta_i\|_0$ where

$$\|.\|_0 = \#(i = 1, \ldots, n; k = 1, \ldots, p|\ \theta_{ik} = 0)$$

is the ℓ_0 norm which measures the parsimony of the displacements of points. Thus, a new optimization problem called (P_1) is given by:

$$(\mathrm{P}_1) : \min_{\theta_1,\ldots,\theta_n \in \mathbb{R}^p} \mathbb{E}(\Delta) + \eta \sum_{i=1}^{n} \|\theta_i\|_0,$$

with η is a positive regularization parameter to be chosen. It controls the trade-off between the minimization of the expectation of the error and the use of a parsimonious number of displacements.

3.2.1 Choice of regularization parameter

In different penalization problems as the penalized regression or penalized likelihood methods for high dimensional data analysis [8], the choice of regularization parameter is always crucial to lead good results attached to the problem at hand. Different methods have been introduced to find the good value of this parameter (see [10],[6]). Some practical approaches consist in comparing different models using a sequence of penalization parameter and then choose the best of them using some model selection criteria like Cross-validation (CV) ([2], [20]), Akaike Information Criterion (AIC) [1] and Bayes Information Criterion (BIC) [19].

In our model, the choice of the value of η is related to the number of displacements. With the same practical concept as the approaches presented in the literature, we want to solve the optimization problem (P_1) by taking different values of η. However, our problem is related to the number of displacements so we choose a value of η that takes into account the number of points that must be modified in our data to fit the references distances. This number of points can be computed from the data or fixed by the number of displacements that we want to perform. So, the chosen number of displacements can be taken by two ways:

1- through the posed problem,

2- using the data.

Obviously, first way is trivial. Indeed, it is sufficient an user or a company choose a fixed number of displacements that wish perform to find the desirable solution. Accordingly, fixing the number of displacements can be interesting to companies because in some cases a displacement can be difficult and expensive therefore it is suitable for them to fix at the beginning the number of displacements. For the second way, the number of displacements can be calculated using the data. Therefore, a criterion of points selection defined below is used to choose the number of displacements.

3.2.1.1 Criterion for selection points

The number of displacements is related to the number of points that are misplaced in their initial configuration and need movements to fit their reference distances. For that, we have developed a simple criterion based on the error calculated on the initial data before data modification. This criterion for selection of the points is denoted ρ_i.
Indeed, for $i = 1, \ldots, n$ and $j = i+1, \ldots, n$, we calculate the following difference:

$$r_{ij} = (d_{ij} - a\|X_i - X_j\|_2)^2.$$

Note that r_{ij} is equivalent to e_{ij} with $L_i = L_j = 0$.
Then, for each $i = 1, \ldots, n$, the criterion for selection points is defined as:

$$\rho_i = \frac{\sum\limits_{m=1, m\neq i}^{n} r_{im}}{\sum\limits_{1\leq i<j\leq n}^{n} r_{ij}}.$$

The values of ρ_i are between 0 and 1 so, for fixed value $\varrho \in [0, 1]$ which is chosen through the value of ρ_i:

- if $\rho_i \leq \varrho$, then i is considered as correctly placed point,
- else, i is considered as misplaced point.

Now, in order to verify the interest of the modification of coordinates so as to approximate the distances, we want to perform a statistical test on the displacement vectors $(\theta_1^*, \ldots, \theta_n^*)$.

3.2.2 Statistical test for the displacement vectors $(\theta_1^*, \ldots, \theta_n^*)$

In this section, we want to present a statistical test for the displacement vectors θ_i for all $i = 1, \ldots, n$. This test is based on the hypothesis of displacements significance. Recall the error:

$$\Delta = \sum_{1\leq i<j\leq n} (d_{ij} - a\|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|_2)^2 . \tag{3.1}$$

We note Δ_0 the initial error given by:

$$\Delta_0 = \sum_{1\leq i<j\leq n} (d_{ij} - a\|X_i + \varepsilon_i - X_j - \varepsilon_j\|_2)^2.$$

The two hypothesis of the statistical test are:

$$\begin{cases} & \text{For } (\theta_1^*, \ldots, \theta_n^*) \text{ such that } d_{ij} = a\|X_i + \theta_i - X_j - \theta_j\|_2, \text{ for all } (i,j), \\ (\mathcal{H}_0): & \text{the initial error } \Delta_0 \text{ and the error } \Delta \text{ calculated from} \\ & \text{the vectors } (\theta_1^*, \ldots, \theta_n^*) \text{ have the same distribution.} \\ (\mathcal{H}_1): & \text{The initial error } \Delta_0 \text{ and the error } \Delta \text{ calculated from the vectors} \\ & (\theta_1^*, \ldots, \theta_n^*) \text{have not the same distribution.} \end{cases}$$

The error Δ is the test statistic and the decision rule is the following:

$$\text{Rejection of } (\mathcal{H}_0) \Leftrightarrow \mathbb{P}_{\mathcal{H}_0}[\text{Reject } (\mathcal{H}_0)] \leq \alpha \Leftrightarrow \mathbb{P}_{\mathcal{H}_0}[\Delta \geq \Delta_c] \leq \alpha\cdot$$

To perform this test, we use the Bienaymé-Tchebychev inequality:

$$\forall \gamma > 0,\ \ \mathbb{P}[|\Delta - \mathbb{E}(\Delta)| \geq \gamma] \leq \frac{\mathbb{V}ar(\Delta)}{\gamma^2}\cdot$$

Moreover, we suppose that the random effect is injected in the observation so instead of observing X_i we observe $X_i + \varepsilon_i$. Then, by choosing $\gamma = |\Delta_0 - \mathbb{E}(\Delta)|$, the ratio $\dfrac{\mathbb{V}ar(\Delta)}{\gamma^2}$ can be considered as p-value. So, if it is small than α then we reject the null hypothesis $(\mathcal{H}_0)$ with α is the error of type I.

Computation of expectation and variance of error Δ are done in Section 3.4. Under the hypothesis $(\mathcal{H}_0)$ it is sufficient to replace $\|X_i + \theta_i - X_j - \theta_j\|_2$ by d_{ij}.

3.2.3 The optimization problem

Once hypothesis $(\mathcal{H}_0)$ is rejected, the vectors $(\theta_1^*, \ldots, \theta_n^*)$ can be calculated by solving problem (P_1).

Using the results of Section 3.4, the expectation of the error Δ has been calculated from the expectation of the non-central chi-squared and non-central chi distribution.

Proposition 3.2.1. *The expectation of the error Δ is:*

$$\mathbb{E}(\Delta) = \sum_{1\leq i<j\leq n} \left[d_{ij}^2 + 2a^2\sigma^2(p + \lambda_{ij}^2) - 2\sqrt{\pi}a\sigma d_{ij} \mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right)\right],$$

where $\lambda_{ij} = \frac{1}{\sqrt{2}\sigma}\|X_i + \theta_i - X_j - \theta_j\|_2$ *and* $\mathcal{L}_\nu^\gamma(x)$ *is the generalized Laguerre polynomial [5].*

The optimization problem (P_1) can rewritten as:

$$(\mathrm{P}_1) : \min_{\theta_1,\ldots,\theta_n \in \mathbb{R}^p} a^2\|X_i+\theta_i-X_j-\theta_j\|_2^2 - 2\sqrt{\pi}a\sigma d_{ij}\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\|X_i+\theta_i-X_j-\theta_j\|_2^2}{4\sigma^2}\right)+\eta\sum_{i=1}^{n}\|\theta_i\|_0.$$

3.3 Calculation of $(\theta_1^*, \ldots, \theta_n^*)$ by simulation

In this section, we suppose that the p components of ε_i are dependent or/and not necessarily normally distributed so the application of chi-squared and chi distributions becomes impossible. Therefore, we want to present an algorithm noted Algorithm 1 which allows us to find the optimal vectors $\theta_1^*, \ldots, \theta_n^*$ using Metropolis-Hastings [17].

3.3.1 Simulation tools

Different tools used in algorithm 1 and associated to the generation of vectors $\theta_1, \ldots, \theta_n$ in order to minimize the error Δ are presented in the follow.

3.3.1.1 Identification of misplaced and correctly placed sets

The set of points can be divided into two subsets:

- The first, noted W, having size equal to n_W. This subset contains the points that are correctly placed and should not be moved.
- The second, noted M, having size equal to n_M. This subset contains the points that are misplaced and must be moved.

The criterion for points selection ρ_i presented in Section 3.2.1.1 is used to construct these two subsets.

3.3.1.2 Movement of set M

The subset M contains the misplaced points that must be moved in order to fit the reference distances. In this section, the work is concentrated to find movements for the subset M approaching as possible as the distances calculated after movements to the reference distances. The movements that can be applied to M are translation, scaling and rotation. The scaling movement is not interest in our study as the subsets W and M are in the same scale seen that are derived from the same set of points and the scaling variable a presented in the optimization problem of MDF is kept. Moreover, we suppose that the points inside M are well concentrated so that the rotation movement

can be neglected. That is why we are just interested on the translation movement. The translation of M through a vector $B \in \mathbb{R}^p$ can be shown as the translation of each points in M. So, the translation of a point $j \in M$ is given by: $Y_j + B$ where $Y_j \in \mathbb{R}^p$ is the coordinate vector of point j.

The translation movement is performed in such a way to approach the distances calculated after translation to the distances given by the reference matrix. Thus, find the vector B return to solve the following optimization problem:

$$(\mathcal{P}) : \min_{B\in\mathbb{R}^p} \sum_{i\in W}\sum_{j\in M} \left(d_{ij}^2 - a \parallel X_i - Y_j - B \parallel_2^2\right)^2.$$

In order to simplify the problem $(\mathcal{P})$, we suppose that for all $i \in W$ and $j \in M$, $d_{ij}^2 - a \parallel X_i - Y_j - B \parallel_2^2 \geq 0$ and the problem $(\mathcal{P})$ becomes:

$$(\mathcal{P}_1) \begin{cases} \min\limits_{B\in\mathbb{R}^p} \sum\limits_{i\in W}\sum\limits_{j\in M} \left(d_{ij}^2 - a \parallel X_i - Y_j - B \parallel_2^2\right) \\ \text{s.t} \quad \forall i \in W \text{ and } j \in M,\ d_{ij}^2 - a \parallel X_i - Y_j - B \parallel_2^2 \geq 0. \end{cases}$$

Relaxation of problem $(\mathcal{P}_1)$: The following problem $(\mathcal{P}_2)$ can easily solved and provide a starting point to resolve $(\mathcal{P}_1)$. We have:

$$(\mathcal{P}_2) : \quad \sum_{i\in W}\sum_{j\in M} \left(d_{ij}^2 - a \parallel X_i - Y_j - B \parallel_2^2\right) = 0 \tag{3.2}$$

$$\Leftrightarrow \quad \sum_{i\in W}\sum_{j\in M} a\left(\parallel X_i - Y_j \parallel_2^2 + \parallel B \parallel_2^2 - 2\langle X_i - Y_j, B\rangle\right) - \sum_{i\in W}\sum_{j\in M} d_{ij}^2 = 0$$

$$\Leftrightarrow \quad a \parallel B \parallel_2^2 - 2a\frac{\sum\limits_{i\in W}\sum\limits_{j\in M}\langle X_i - Y_j, B\rangle}{n_W n_M} = \frac{\sum\limits_{i\in W}\sum\limits_{j\in M} d_{ij}^2}{n_W n_M} - a\frac{\sum\limits_{i\in W}\sum\limits_{j\in M}\|X_i - Y_j\|_2^2}{n_W n_M}.$$

Hence,

$$a\left\|B - \frac{\sum\limits_{i\in W}\sum\limits_{j\in M}(X_i - Y_j)}{n_W n_M}\right\|_2^2 = \frac{\sum\limits_{i\in W}\sum\limits_{j\in M} d_{ij}^2}{n_W n_M} - a\frac{\sum\limits_{i\in W}\sum\limits_{j\in M} \parallel X_i - Y_j \parallel_2^2}{n_W n_M} + a\frac{\left(\parallel \sum\limits_{i\in W}\sum\limits_{j\in M}(X_i - Y_j) \parallel_2\right)^2}{n_W^2 n_M^2}.$$

So,

$$\left\|B - \frac{\sum\limits_{i\in W}\sum\limits_{j\in M}(X_i - Y_j)}{n_W n_M}\right\|_2^2 \leq r^2, \tag{3.3}$$

with

$$r^2 = \left| \frac{\sum_{i\in W}\sum_{j\in M} d_{ij}^2}{a\ n_W n_M} - \frac{\sum_{i\in W}\sum_{j\in M} \parallel X_i - Y_j \parallel_2^2}{n_W n_M} + \frac{\left(\parallel \sum_{i\in W}\sum_{j\in M} (X_i - Y_j) \parallel_2 \right)^2}{n_W^2 n_M^2} \right|.$$

Using inequality (3.3), we can conclude that the optimal solution of the vector B belongs to an hypersphere (S) centered in $C = \frac{\sum_{i\in W}\sum_{j\in M}(X_i - Y_j)}{n_W n_M}$ with radius r. As Equation (3.2) is never equal to zero during optimization, so we take it smaller than certain real value. Therefore, the optimal solution is guened to belong to an hypersphere (S^ξ) with same center C as hypersphere (S) but with radius equal to $r \pm \xi$ with ξ a small value in $\mathbb{R}^+$.

We suppose that vector B is uniformly distributed between 0 and B_{max}, so it is necessary to find the maximal value B_{max}. This value is geometrically determined using Figure 3.1. Starting with a null value of vector B, B moves uniformly on the line (d) passing by O (the point where the vector B is null) and C to reach its maximum at the point A, the far intersection between (d) and hypersphere (S^ξ), hence the uniqueness of A.

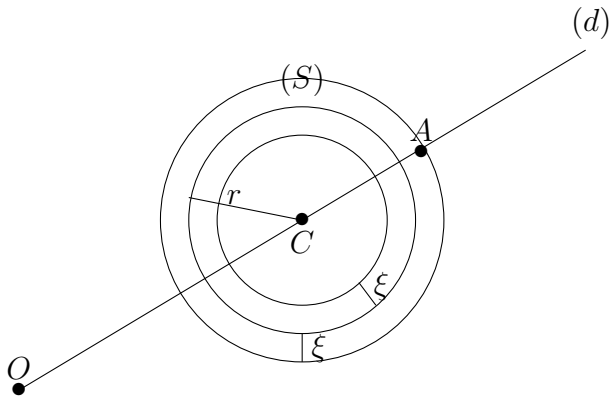


Figure 3.1: Illustration of the determination of vector B in $\mathbb{R}^2$. The maximal solution of B is located at A. The values of vector B moves uniformly on the segment $[OA]$.

To calculate the maximal solution of vector B, it is needed to find the far intersection of the line (d) with hypersphere (S^ξ). The line (d) has as direction vector the vector OC. So, the parametric equation of (d) is equal to:

$$B = (t+1)C\cdot \tag{3.4}$$

Furthermore, we have:

$$\begin{cases} B = (t+1)C \\ \parallel B - C \parallel_2^2 = (r+\xi)^2. \end{cases}$$

The intersection between (d) and (S^{ξ}) gives:

$$\begin{aligned} \| (t+1)C - C \|_2^2 &= (r+\xi)^2 \\ \| tC \|_2^2 &= (r+\xi)^2 \\ t^2 \| C \|_2^2 &= (r+\xi)^2 \\ t &= \pm \frac{r+\xi}{\| C \|_2}\cdot \end{aligned}$$

We are interested by the farthest intersection, thus we take $t = \dfrac{r+\xi}{\| C \|_2}\cdot$ By replacing t in Equation (3.4), we obtain:

$$B_{max} = \left(\frac{r+\xi}{\| C \|_2} + 1 \right) C.$$

The values of B can be proposed uniformly on the segment $[OA]$, so

$$B \rightsquigarrow \mathcal{U}\left(0, B_{max}\right).$$

3.3.1.3 Movement vectors generation

In practice, for all $k = 1, \ldots, n$, we suppose that M contains one point noted l. The choice of this point is made by a multinomial distribution $\mathcal{M}(1, \rho_1, \ldots, \rho_n)$ where ρ_k for $k = 1, \ldots, n$ is as defined in section 3.2.1.1.

At instant t, a point l chosen as misplaced point must occur a movement through the uniform distribution such as $\mathcal{U}\left(0; (\frac{r_l+\xi}{\|C_l\|}+1)C_l\right)$ with C_l and r_l are, respectively, the center and the radius of hypersphere (S_l^{ξ}) obtained by taking $M = \{l\}$. Thus, the movement of the point l is equal to the movement at instant $t-1$ plus the new movement obtained by uniform distribution. Hence, we can write:

$$\theta_l^* = \theta_l^{t-1} + B_l,$$

with B_l is a generation value of the uniform distribution $\mathcal{U}\left(0, B_{max}^l\right)$. Noted that the equation of hypersphere given by Equation (3.3) in each instant depends of misplaced point l.

We note Θ the sequence of n generated vectors in $\mathbb{R}^p$ defined by:

$\Theta = (\theta_1, \ldots, \theta_n)$. Therefore, the passage from θ_l^{t-1} to θ_l^t occurs in a way to move:

$$\Theta^{t-1} = (\theta_1^{t-1}, \ldots, \theta_l^{t-1}, \ldots, \theta_n^{t-1})$$

to

$$\Theta^{t} = (\theta_1^{t-1}, \ldots, \theta_l^{t}, \ldots, \theta_n^{t-1}).$$

3.3.1.4 Proposal distribution

A proposal distribution is needed in the Metropolis-Hastings algorithm defined below. This distribution is constructed by calculating the probability to pass from Θ^{t-1} to a new generate value of Θ denoted Θ^* and it is equal to the probability to choose a point l multiplied by the probability of the movement of this point. So, the proposal distribution

noted q is given by:

$$q(\Theta^{t-1} \longrightarrow \Theta^*) = \rho_l \times \frac{1}{\frac{r_l+\xi}{\|C_l\|}+1}$$

with l is the chosen point.

We can easily see that this proposal distribution is a probability density function as $\sum_{i=1}^{n} \rho_i = 1$.

3.3.2 Calculation of $(\theta_1^*, \ldots, \theta_n^*)$ using Metropolis-Hastings algorithm

We consider that the component of vector ε_i are dependent such that $\varepsilon_i \rightsquigarrow \mathcal{N}_p(\mathbf{0}, \Sigma)$, with Σ is the covariance matrix.

The Metropolis-Hastings algorithm allows us to build a Markov chain with a desired stationary distribution [17],[9]. The proposal distribution here is related to the choice of vectors θ_i for $i = 1, \ldots, n$ and it is given in paragraph 3.3.1.4. The target distribution is given by:

$$\pi(\Theta, \varepsilon) \propto \exp\left(\frac{-E(\Theta)}{T}\right) \cdot h(\varepsilon)$$

where E is an application given by:

$$\begin{array}{rccc} E: & \mathcal{M}_{n\times p} & \longmapsto & \mathbb{R} \\ & \Theta = (\theta_1, \ldots, \theta_n) & \longmapsto & E(\Theta) = \displaystyle\sum_{1\le i<j\le n} \left(d_{ij} - a\|X_i + \theta_i - X_j - \theta_j\|_2\right)^2, \end{array}$$

and h is the density function of the normal distribution $\mathcal{N}_p(\mathbf{0}, \Sigma)$. The variable T is the temperature parameter, to be fixed according to the value range of E.

Algorithm 1 describes the Metropolis-Hastings algorithm of our method.

Remark: The error ε_i can be distributed through any other distribution other than the Gaussian distribution, so it is sufficient to generate the vector ε_i using this distribution instead the normal distribution in Algorithm 1.

3.4 Calculation of the expectation and the variance of the error Δ

The statistical test presented in Section 3.2.2 needs the computation of the expectation and the variance of the error Δ. So first of all, we want to give fives lemmas used in the computation and then the computation of these values.

3.4.1 Fives Lemmas used in the calculation

We have:

$$e_{ij} \;=\; \left(d_{ij} - a\|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|_2\right)^2.$$

Algorithm 1

Initialization: $\Theta_0 = (\theta_1|\dots|\theta_n) = (\mathbf{0}|\dots|\mathbf{0})$.

Calculate the ratios $\rho_1, \dots, \rho_n$.

for $t = 1$ to N_1 **do**

Generate a point l using multinomial distribution $\mathcal{M}(1, \rho_1, \dots, \rho_n)$.

Generate a vector B_l using the uniform distribution $\mathcal{U}(0, B^l_{max})$ with $B^l_{max} = \left(\frac{r_l+\xi}{\|C_l\|_2} + 1\right) C_l$.

Generate the vector $\theta^*_l = \theta^{t-1}_l + B_l$ and for all $i \in [1, \dots, n] - \{l\}$ take the vectors θ^*_i equal to θ^{t-1}_i.

Generate the vectors ε^*_i using normal distribution $\mathcal{N}(\mathbf{0}, \Sigma)$ for $i = 1, \dots, n$.

Calculate $\alpha = \min\{1, \dfrac{\exp\left(\frac{-E(\Theta^*)}{T}\right) h(\varepsilon^*) q(\Theta^* \to \Theta^{t-1})}{\exp\left(\frac{-E(\Theta^{t-1})}{T}\right) h(\varepsilon^{t-1}) q(\Theta^{t-1} \to \Theta^*)}\}$.

Generate $u \sim \mathcal{U}(0, 1)$.

if $u \leq \alpha$ **then**

$\Theta^t = \Theta^*$

else

$\Theta^t = \Theta^{t-1}$.

end if

end for

Choose Θ that gives the minimum value of error Δ

By developing the expression of e_{ij}, we obtain:

$$e_{ij} = d^2_{ij} + a^2\|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|^2_2 - 2\ a\ d_{ij}\|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|_2 \quad (3.5)$$

We want to present two lemmas that will help us in the calculation of expectation and variance of Δ.

Lemma 3.1. *Let N_{ij} be a random variable defined by:*

$$N_{ij} = \sum_{k=1}^{p} \left(\varepsilon_{ik} - \varepsilon_{jk}\right)^2, \forall 1 \leq i < j \leq n \quad (3.6)$$

where the p components of vectors ε_i, for all $i = 1, \dots, n$ are identically independent random variables and normally distributed such that $\varepsilon_{ik} \rightsquigarrow \mathcal{N}(0, \sigma^2)$ for $k = 1, \dots, p$. Then, we have:

$$\mathbb{E}(N_{ij}) = 2\sigma^2 p \text{ et } \mathbb{V}ar(N_{ij}) = 8\sigma^4 p.$$

Proof. So that, $\varepsilon_{ik} \rightsquigarrow \mathcal{N}(0, \sigma^2)$ and the vectors ε_i and ε_j are independents, we have $\varepsilon_{ik} - \varepsilon_{jk} \rightsquigarrow \mathcal{N}(0, 2\sigma^2)$. Thus, $\sum_{k=1}^{p} \left(\dfrac{\varepsilon_{ik} - \varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2 \rightsquigarrow \chi^2_p$, and consequently:

$$\mathbb{E}\left(\sum_{k=1}^{p} \left(\frac{\varepsilon_{ik} - \varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2\right) = p \text{ et } \mathbb{V}ar\left(\sum_{k=1}^{p} \left(\frac{\varepsilon_{ik} - \varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2\right) = 2p.$$

So, we obtain:

$$\begin{aligned}\mathbb{E}\left(N_{ij}\right) &= 2\sigma^2 p\\ \mathbb{V}ar\left(N_{ij}\right) &= 8\sigma^4 p.\end{aligned}$$

□

Lemma 3.2. *Let A_{ij} be a random variable defined by:*

$$A_{ij} = \|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|_2 \tag{3.7}$$

with $\varepsilon_{ik} \rightsquigarrow \mathcal{N}(0,\sigma^2)$. Then, we have:

$$\begin{aligned}\mathbb{E}(A_{ij}) &= \sqrt{2}\sigma\mu_{ij}\\ \mathbb{V}ar(A_{ij}) &= 2\sigma^2(p+\lambda_{ij}^2-\mu_{ij}^2)\\ \mathbb{E}(A_{ij}^2) &= 2\sigma^2(p+\lambda_{ij}^2)\\ \mathbb{V}ar(A_{ij})^2) &= 8\sigma^4(p+2\lambda_{ij}^2)\\ \mathbb{E}\left(A_{ij}^3\right) &= 6\sigma^3\sqrt{\pi}\mathcal{L}_{\frac{3}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right)\\ \mathbb{E}\left(A_{ij}^4\right) &= 4\sigma^4(p+\lambda_{ij}^2)^2+8\sigma^4(p+2\lambda_{ij}^2)\end{aligned}$$

with $\mu_{ij} = \sqrt{\frac{\pi}{2}}\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right)$, $\lambda_{ij} = \sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}-x_{jk}-\theta_{jk}}{\sqrt{2}\sigma}\right)^2}$ and $\mathcal{L}_\nu^{(\alpha)}(x)$ is the generalized Laguerre polynomial.

Proof. A_{ij} is a random variable defined by:

$$A_{ij} = \|X_i + \theta_i + \varepsilon_i - X_j - \theta_j - \varepsilon_j\|_2 = \sqrt{\sum_{k=1}^{p}\left(x_{ik}+\theta_{ik}+\varepsilon_{ik}-x_{jk}-\theta_{jk}-\varepsilon_{jk}\right)^2}.$$

The random variable $x_{ik}+\theta_{ik}+\varepsilon_{ik}-x_{jk}-\theta_{jk}-\varepsilon_{jk}$ is normally distributed such that $\mathcal{N}(x_{ik}+\theta_{ik}-x_{jk}-\theta_{jk},2\sigma^2)$ which implies that the random variable

$$\sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}+\varepsilon_{ik}-x_{jk}-\theta_{jk}-\varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2}$$

is distributed according to the non-central chi-squared distribution with p degrees of freedom and λ_{ij} the non-centrality parameter that is related to the mean of the random variable by: $\lambda_{ij} = \sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}-x_{jk}-\theta_{jk}}{\sqrt{2}\sigma}\right)^2}$.

Then, we obtain:

$$\mathbb{E}\left(\sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}+\varepsilon_{ki}-x_{jk}-\theta_{jk}-\varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2}\right) = \frac{1}{\sqrt{2}\sigma}\mathbb{E}(A_{ij}).$$

Recall that the expectation of non-central chi distribution $\chi_p(\lambda_{ij})$ is given by:

$$\sqrt{\frac{\pi}{2}}\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right),$$

where $\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}(-\frac{\lambda_{ij}^2}{2})$ is the generalized Laguerre polynomial [5].

We note

$$\mu_{ij} = \sqrt{\frac{\pi}{2}}\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right).$$

Then, we have

$$\mathbb{E}(A_{ij}) = \sqrt{2}\sigma\mu_{ij}.$$

Moreover, we have:

$$\mathbb{V}ar\left(\sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}+\varepsilon_{ik}-x_{jk}-\theta_{jk}-\varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2}\right) = \frac{1}{2\sigma^2}\mathbb{V}ar(A_{ij}).$$

Recall that the variance of non-central chi distribution $\chi_p(\lambda_{ij})$ is given by:

$$p + \lambda_{ij}^2 - \mu_{ij}^2$$

that gives:

$$\mathbb{V}ar(A_{ij}) = 2\sigma^2(p + \lambda_{ij}^2 - \mu_{ij}^2).$$

Concerning the calculation of moments of order 3 and 4, we have:

$$\mathbb{E}\left(\sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}+\varepsilon_{ik}-x_{jk}-\theta_{jk}-\varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2}\right)^3 = \frac{1}{2\sqrt{2}\sigma^3}\mathbb{E}(A_{ij}^3)$$

and

$$\mathbb{E}\left(\sqrt{\sum_{k=1}^{p}\left(\frac{x_{ik}+\theta_{ik}+\varepsilon_{ik}-x_{jk}-\theta_{jk}-\varepsilon_{jk}}{\sqrt{2}\sigma}\right)^2}\right)^4 = \frac{1}{4\sigma^4}\mathbb{E}(A_{ij}^4).$$

For a non-central chi distribution, the moments 3 and 4 are given respectively by:

$$3\sqrt{\frac{\pi}{2}}\mathcal{L}_{\frac{3}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right) \text{ and } (p+\lambda_{ij}^2)^2 + 2(p+2(\lambda_{ij}^2)$$

We obtain then:

$$\mathbb{E}(A_{ij}^3) = 6\sigma^3\sqrt{\pi}\mathcal{L}_{\frac{3}{2}}^{\frac{p}{2}-1}(-\frac{\lambda_{ij}^2}{2})$$

and

$$\mathbb{E}(A_{ij}^4) = 4\sigma^4(p+\lambda_{ij}^2)^2 + 8\sigma^4(p+2\lambda_{ij}^2)$$

As ${A^2}_{ij}$ is a random variable distributed according to the non-central chi-squared distribution and having p degrees of freedom and λ_{ij}^2 non-centrality parameter, we obtain:

$$\mathbb{E}({A^2}_{ij}) = 2\sigma^2(p+\lambda_{ij}^2) \qquad \text{and} \qquad \mathbb{V}ar({A^2}_{ij}) = 8\sigma^4(p+2\lambda_{ij}^2).$$

□

Lemma 3.3. *An upper bound of* $\mathbb{E}(A_{ij}A_{ij'})$ *and* $\mathbb{E}(A_{ij}^2A_{ij'}^2)$ *are given by:*

$$\mathbb{E}(A_{ij}A_{ij'}) \leq 2\sigma^2\sqrt{(p+\lambda_{ij}^2)(p+\lambda_{ij'}^2)},$$

$$\mathbb{E}(A_{ij}^2A_{ij'}^2) \leq 4\sigma^4\sqrt{\left[(p+\lambda_{ij}^2)^2+2(p+2\lambda_{ij}^2)\right]\left[(p+\lambda_{ij'}^2)^2+2(p+2\lambda_{ij'}^2)\right]}.$$

Proof. The variables A_{ij} and $A_{ij'}$ are two dependent random variables.
Using Cauchy-Schwartz inequality, we can write:

$$\mathbb{E}(A_{ij}A_{ij'}) \leq \sqrt{\mathbb{E}(A_{ij}^2)\mathbb{E}(A_{ij'}^2)}$$

Using Lemma 3.2, we obtain:

$$\mathbb{E}(A_{ij}A_{i'j'}) \leq 2\sigma^2\sqrt{(p+\lambda_{ij}^2)(p+\lambda_{ij'}^2)}\cdot$$

Moreover,

$$\mathbb{E}(A_{ij}^2A_{ij'}^2) \leq \sqrt{\mathbb{E}(A_{ij}^4)\mathbb{E}(A_{ij'}^4)}\cdot$$

And from lemma 3.2, we obtain:

$$\mathbb{E}(A_{ij}^2A_{ij'}^2) \leq 4\sigma^4\sqrt{\left[(p+\lambda_{ij}^2)^2+2(p+2\lambda_{ij}^2)\right]\left[(p+\lambda_{ij'}^2)^2+2(p+2\lambda_{ij'}^2)\right]}\cdot$$

□

Lemma 3.4. *A lower bounds of* $\mathbb{V}ar(A_{ij}+A_{ij'})$, $cov(A_{ij},A_{ij'})$ *and* $\mathbb{E}(A_{ij}A_{ij'})$ *are given by:*

$$\mathbb{V}ar(A_{ij}+A_{ij'}) \geq 2\sigma^2\left(p+\lambda_{jj'}^2-(\mu_{ij}+\mu_{ij'})^2\right),$$

$$cov(A_{ij},A_{ij'}) \geq -\sigma^2\left(p+2\mu_{ij}\mu_{ij'}+\lambda_{ij}^2+\lambda_{ij'}^2-\lambda_{jj'}^2\right),$$

$$\mathbb{E}(A_{ij}A_{ij'}) \geq -\sigma^2\left(p+\lambda_{ij}^2+\lambda_{ij'}^2-\lambda_{jj'}^2\right)\cdot$$

Proof. We have:

$$\begin{aligned} A_{ij}+A_{ij'} &= \|X_i+\theta_i+\varepsilon_i-X_j-\theta_j-\varepsilon_j\|+\|X_i+\theta_i+\varepsilon_i-X_{j'}-\theta_{j'}-\varepsilon_{j'}\| \\ &\geq \|X_j+\theta_j+\varepsilon_j-X_{j'}-\theta_{j'}-\varepsilon_{j'}\| = A_{jj'}\cdot \end{aligned} \tag{3.8}$$

The variance of $A_{ij}+A_{ij'}$ is given by:

$$\mathbb{V}ar(A_{ij}+A_{ij'}) = \mathbb{E}(A_{ij}+A_{ij'})^2-\mathbb{E}^2(A_{ij}+A_{ij'})\cdot$$

Using inequality 3.8 and as the variables A_{ij} are positives for all couples (i,j), we have $(A_{ij}+A_{ij'})^2 \geq A_{jj'}^2$ and then:

$$\mathbb{E}\left((A_{ij}+A_{ij'})^2\right) \geq \mathbb{E}(A_{jj'}^2) = 2\sigma^2(p+\lambda_{jj'}^2)\cdot$$

Hence, we have:

$$\begin{aligned} \mathbb{V}ar(A_{ij}+A_{ij'}) &\geq 2\sigma^2(p+\lambda_{jj'}^2)-(\mathbb{E}(A_{ij})+\mathbb{E}(A_{ij'}))^2 \\ &\geq 2\sigma^2(p+\lambda_{jj'}^2)-2\sigma^2(\mu_{ij}+\mu_{ij'})^2 \\ &\geq 2\sigma^2\left(p+\lambda_{jj'}^2-(\mu_{ij}+\mu_{ij'})^2\right)\cdot \end{aligned}$$

To obtain the lower bound of $cov(A_{ij},A_{ij'})$, we use the definition:

$$cov(A_{ij},A_{ij'}) = \frac{1}{2}\left[\mathbb{V}ar(A_{ij}+A_{ij'})-\mathbb{V}ar(A_{ij})-\mathbb{V}ar(A_{ij'})\right]\cdot$$

Then, using Lemma 3.2 and the above result, we obtain:

$$\begin{aligned} cov(A_{ij},A_{ij'}) &\geq \sigma^2(p+\lambda_{jj'}^2-(\mu_{ij}+\mu_{ij'})^2)-\sigma^2(p+\lambda_{ij}^2-\mu_{ij}^2)-\sigma^2(p+\lambda_{ij'}^2-\mu_{ij'}^2) \\ &\geq -\sigma^2\left(p+2\mu_{ij}\mu_{ij'}+\lambda_{ij}^2+\lambda_{ij'}^2-\lambda_{jj'}^2\right)\cdot \end{aligned} \tag{3.9}$$

Concerning the lower bound of $\mathbb{E}(A_{ij}A_{ij'})$, we have:

$$\begin{aligned} \mathbb{E}(A_{ij}A_{ij'}) &= cov(A_{ij},A_{ij'})+\mathbb{E}(A_{ij})\mathbb{E}(A_{ij'}) \\ &\geq -\sigma^2\left(p+2\mu_{ij}\mu_{ij'}+\lambda_{ij}^2+\lambda_{ij'}^2-\lambda_{jj'}^2\right)+2\sigma^2\mu_{ij}\mu_{ij'} \\ &\geq -\sigma^2\left(p+\lambda_{ij}^2+\lambda_{ij'}^2-\lambda_{jj'}^2\right)\cdot \end{aligned}$$

□

Lemma 3.5. *A lower bounds of* $\mathbb{E}(A_{ij}^2 A_{ij'})$ *and* $\mathbb{E}(A_{ij'}^2 A_{ij})$ *are given by:*

$$\mathbb{E}(A_{ij}^2 A_{ij'}) \geq -\sigma^2\left(p + \lambda_{ij}^2 + \lambda_{ij'}^2 - \lambda_{jj'}^2\right) - \sqrt{2}\sigma\mu_{ij'},$$
$$\mathbb{E}(A_{ij'}^2 A_{ij}) \geq -\sigma^2\left(p + \lambda_{ij}^2 + \lambda_{ij'}^2 - \lambda_{jj'}^2\right) - \sqrt{2}\sigma\mu_{ij}.$$

Proof.

$$\begin{aligned} A_{ij}^2 A_{ij'} &\geq (A_{ij} - 1)A_{ij'} \text{ as } A_{ij}^2 \geq A_{ij} - 1 \\ &\geq A_{ij}A_{ij'} - A_{ij'}. \end{aligned}$$

Thus,

$$\mathbb{E}\left(A_{ij}^2 A_{ij'}\right) \geq \mathbb{E}(A_{ij}A_{ij'}) - \mathbb{E}(A_{ij'}).$$

Lemma 3.4 leads:

$$\mathbb{E}(A_{ij}^2 A_{ij'}) \geq -\sigma^2\left(p + \lambda_{ij}^2 + \lambda_{ij'}^2 - \lambda_{jj'}^2\right) - \sqrt{2}\sigma\mu_{ij'}.$$

Similarly, we obtain:

$$\mathbb{E}(A_{ij'}^2 A_{ij}) \geq -\sigma^2\left(p + \lambda_{ij'}^2 + \lambda_{ij}^2 - \lambda_{jj'}^2\right) - \sqrt{2}\sigma\mu_{ij}.$$

□

3.5 Calculation of the expectation value of error Δ

Using equations (3.5) and (3.7), we have:

$$e_{ij} = d_{ij}^2 + a^2 A_{ij}^2 - 2ad_{ij}A_{ij}. \tag{3.10}$$

We suppose that the p components of vectors ε_i, for all $i = 1, \ldots, n$ are identically independent random variables and normally distributed. Using Lemma 3.2, we obtain:

$$\mathbb{E}(e_{ij}) = d_{ij}^2 + 2a^2\sigma^2(p + \lambda_{ij}^2) - 2\sqrt{\pi}a\sigma d_{ij}\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right).$$

where $\lambda_{ij} = \dfrac{1}{\sqrt{2}\sigma}\|X_i + \theta_i - X_j - \theta_j\|_2$.

Hence, the expectation of error Δ is equal to:

$$\mathbb{E}(\Delta) = \sum_{1\leq i<j\leq n}\left[d_{ij}^2 + 2a^2\sigma^2(p + \lambda_{ij}^2) - 2\sqrt{\pi}a\sigma d_{ij}\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right)\right]. \tag{3.11}$$

3.6 Calculation of variance value of error Δ

The variance of Δ is given by:

$$\begin{aligned} \mathbb{V}ar(\Delta) &= \mathbb{V}ar\Big(\sum_{1\leq i<j\leq n} e_{ij}\Big) \\ &= \sum_{1\leq i<j\leq n}\mathbb{V}ar(e_{ij}) + \sum_{\substack{1\leq i<j\leq n \\ 1\leq i'<j'\leq n}} cov(e_{ij}; e_{i'j'}) \end{aligned}$$

As $cov(e_{ij}, e_{i'j'}) = 0$, if $(i,j) \cap (i',j') = \emptyset$ we obtain:

$$\mathbb{V}ar(\Delta) = \sum_{1\leq i<j\leq n}\mathbb{V}ar(e_{ij}) + 2\sum_{1\leq i<j<j'\leq n} cov(e_{ij}, e_{ij'}). \tag{3.12}$$

To calculate $\mathbb{V}ar(\Delta)$ it is necessary to calculate $\mathbb{V}ar(e_{ij})$ and $cov(e_{ij}; e_{ij'})$ for all couples (i, j) and (i, j') with $1 \leq i < j < j' \leq n$.

3.6.1 Calculation of $\mathbb{V}ar(e_{ij})$

We have from Equation (3.10):

$$e_{ij} = d_{ij}^2 + a^2 A_{ij}^2 - 2ad_{ij}A_{ij}\cdot$$

The definition of variance is:

$$\mathbb{V}ar(e_{ij}) = \mathbb{E}(e_{ij}^2) - (\mathbb{E}(e_{ij}))^2\cdot \tag{3.13}$$

Let begin by the calculation of $\mathbb{E}(e_{ij}^2)$.

$$\begin{aligned} e_{ij}^2 &= \left(d_{ij}^2 + a^2 A_{ij}^2 - 2ad_{ij}A_{ij}\right)^2 \\ &= d_{ij}^4 + a^4 A_{ij}^4 + 6a^2 d_{ij}^2 A_{ij}^2 - 4ad_{ij}^3 A_{ij} - 4a^3 d_{ij} A_{ij}^3\cdot \end{aligned}$$

The expectation of e_{ij}^2 is then given by:

$$\mathbb{E}(e_{ij}^2) = d_{ij}^4 + a^4\mathbb{E}(A_{ij}^4) + 6a^2 d_{ij}^2\mathbb{E}(A_{ij}^2) - 4ad_{ij}^3\mathbb{E}(A_{ij}) - 4a^3 d_{ij}\mathbb{E}(A_{ij}^3)\cdot \tag{3.14}$$

Using lemma 3.2, we can obtain all the terms of the moments presented in Eequation (3.14). Then, we obtain the variance by replacing each term with their value in Equation (3.13) and we obtain:

$$\begin{aligned} \mathbb{V}ar(e_{ij}) = \; & a^4\mathbb{E}(A_{ij}^4) + 4a^2 d_{ij}^2\mathbb{E}(A_{ij}^2) - 4a^3 d_{ij}\mathbb{E}(A_{ij}^3) - a^4(\mathbb{E}(A_{ij}^2))^2 - 4a^2 d_{ij}^2(\mathbb{E}(A_{ij}))^2 \\ & + 4a^3 d_{ij}\mathbb{E}(A_{ij}^2)\mathbb{E}(A_{ij})\cdot \end{aligned}$$

3.6.2 Calculation of $cov(e_{ij}, e_{ij'})$

Now, we want to calculate $cov(e_{ij}, e_{ij'})$. The definition of the covariance is given by:

$$cov(e_{ij}, e_{ij'}) = \mathbb{E}(e_{ij}e_{ij'}) - \mathbb{E}(e_{ij})\mathbb{E}(e_{ij'})\cdot \tag{3.15}$$

To calculate the expectation $\mathbb{E}(e_{ij}e_{ij'})$, we firstly calculate $e_{ij}e_{ij'}$:

$$\begin{aligned} e_{ij}e_{ij'} &= \left(d_{ij}^2 + a^2 A_{ij}^2 - 2ad_{ij}A_{ij}\right)\left(d_{ij'}^2 + a^2 A_{ij'}^2 - 2ad_{ij'}A_{ij'}\right) \\ &= d_{ij}^2 d_{ij'}^2 + a^2 d_{ij}^2 A_{ij'}^2 - 2ad_{ij}^2 d_{ij'}A_{ij'} + a^2 d_{ij'}^2 A_{ij}^2 + a^4 A_{ij}^2 A_{ij'}^2 \\ &\quad -2a^3 d_{ij'}A_{ij}^2 A_{ij'} - 2ad_{ij}d_{ij'}^2 A_{ij} - 2a^3 d_{ij}A_{ij}A_{ij'}^2 + 4a^2 d_{ij}d_{ij'}A_{ij}A_{ij'}\cdot \end{aligned}$$

Passing to the expectation, we obtain:

$$\begin{aligned} \mathbb{E}(e_{ij}e_{ij'}) = \; & d_{ij}^2 d_{ij'}^2 + a^2 d_{ij}^2\mathbb{E}(A_{ij'}^2) - 2ad_{ij}^2 d_{ij'}\mathbb{E}(A_{ij'}) + a^2 d_{ij'}^2\mathbb{E}(A_{ij}^2) + a^4\mathbb{E}(A_{ij}^2 A_{ij'}^2) \\ & -2a^3 d_{ij'}\mathbb{E}(A_{ij}^2 A_{ij'}) - 2ad_{ij}d_{ij'}^2\mathbb{E}(A_{ij}) - 2a^3 d_{ij}\mathbb{E}(A_{ij}A_{ij'}^2) + 4a^2 d_{ij}d_{ij'}\mathbb{E}(A_{ij}A_{ij'})\cdot \end{aligned} \tag{3.16}$$

Using the five lemmas presented in section 3.4.1, we can bound $\mathbb{E}(e_{ij}e_{ij'})$ in a way to obtain an upper bound of the covariance $cov(e_{ij}e_{ij'})$. We note $B_{ijj'}$ the upper bound of $\mathbb{E}(e_{ij}e_{ij'})$. So, returning to Equation (3.12), we obtain:

$$\mathbb{V}ar(\Delta) \leq \sum_{1\leq i<j\leq n}\left(\mathbb{E}(e_{ij}^2) - \mathbb{E}(e_{ij})^2\right) + 2\sum_{1\leq i<j<j'\leq n} B_{ijj'} - 2\sum_{1\leq i<j<j'\leq n}\mathbb{E}(e_{ij})\mathbb{E}(e_{ij'})\cdot$$

3.7 Application

This random model of multidimensional fitting has been applied in the sensometrics domain. This relatively young domain concerns the analysis of data from sensory science in order to develop a product by linking sensory attributes to ingredients, benefits, values and emotional elements of the brand to design products that meet the sensory quality preferences of sensory-based consumer segments [15]. This analysis of product characteristics among consumers gives an overview of the positive and negative aspects of products and aid the companies to better meet consumer tastes. So, the problem here is to fit the consumers scores to the product configuration given by the experts in order to find the ideal sensory profile of a product. Thus, two matrices are at disposal, one contains the consumer scores of products and the second the sensory profile of products given by the experts.
Several modelling techniques have been applied in sensory analysis domain like preference mapping which is the must popular of them. They can be divided into two methods: internal and external analysis [13]. These methods have as objective to visually assess the relationship between the product space and patterns of preference [12]. In our application, we want to use the random model of multidimensional fitting to match as well as possible the sensory profile to the consumers preference of a product. White corn tortilla chips and muscadine grape juice data sets are used in our application.

3.7.1 Data description

White corn tortilla chips data set has been studied in [15] where 80 consumers rated 11 commercially available toasted white corn tortilla chip products for overall liking, appearance liking, and flavor liking. The names of these 11 tortilla chip products and their labels are given in the Table 3.1. Moreover, a group of 9 Spectrum trained panelists evaluated apperance, flavor and texture attributes of tortilla chips using the Spectrum Method [13] (Sensory Spectrum Inc., Chantham, NJ, U.S.A.). This data set is available at "`http://www.sensometric.org/datasets`" and it is composed from consumers notes table and panelists notes table. The first table is constructed after asked each consumer to evaluate liking, appearance, flavor and texture of each tortilla chips sample on a 9-point dedonic scale and the saltiness on 5-point 'JustAboutRight' (JAR) (for more information about the scale, visit "`http://www.sensorysociety.org/knowledge/sspwiki/Pages/The209-point20Hedonic20Scale.aspx`"). The second table is obtained after the evaluation of the 9 panelists for flavor, texture and appearance attributes of all the chips and after that the calculation of the average score for each attribute. The total number of attributes studied in panelists notes table is 37, we note some of them: sweet, salt, sour, lime, astringent, grain complex, toasted corn, raw corn, masa, toasted grain...
The application of our method requires a target and reference matrices. The target matrix

is given by the panelists notes table, so the dimension of this matrix is 11×37 and the reference matrix is a matrix of dimension 11×11 and contains the Euclidean distances between the different tortilla chip samples calculated using the consumers notes table.

Muscadine grape juice data set is well studied in [16], and it is composed from the scores of 61 consumers and the average score for 15 attributes given by 9 panelists. This data is available at "`http://www.sensometric.org/datasets`". Consumers evaluated 10 muscadine grape juices for overall impression, appearance, aroma, color, and flavor. The name of the 10 studied muscadine grape cultivars are given in the Table 3.2. These 10 juices are examined for aroma, basic tastes, aromatics, feeling factors by the group of Sensory Spectrum trained panelists. Likewise to white corn tortilla chips data set, this data set is composed from two tables: consumers notes table and panelists notes table. The first table contains the consumers evaluation of overall impression, appearance, aroma, color and flavor on the 9-point hedonic scale and the second one contains the average score for each attribute after evaluation of the 9 panelists for the basic taste, aromatics and feeling factors attributes for all muscadine juices.

The target matrix here is a matrix of dimension 10×15 constructed by the average score given by the panelists and the reference matrix is a matrix of dimension 10×10 constructed by the Euclidean distances between the consumers scores for the different cultivars of muscadine grape juices. To quote some of the studied attributes: sweet, sour, cooked muscadine, cooked grape, musty, green unripe, floral apple/pear, fermented ...

Tortilla Chip names	abb.
Best Yet White Corn	BYW
Green Mountain Gringo	GMG
Guy's Restaurant Rounds	GUY
Medallion White Corn	MED
Mission Strips	MIS
Mission Triangle	MIT
Oak Creek Farms-White Corn	OAK
Santita's	SAN
Tostito's Bite Size	TOB
Tom's White Corn	TOM
Tostito's Restaurant Style	TOR

Table 3.1: White corn tortilla chip product names and labels

Muscadine juice names	abb.
Black Beauty	BB
Carlos	CA
Granny Val	GV
Ison	IS
Nestitt	ME
Commercial Red	CR
Commercial White	CW
Southern Home	SH
Summit	SUM
Supreme	SUP

Table 3.2: Muscadine grape juice names and labels

3.7.2 Experimental setup

Random model of multidimensional fitting method is applied in the independent and dependent cases of the components of vectors ε_i for $i = 1, \ldots, n$. The presence of Laguerre polynomial $\mathcal{L}_{\frac{1}{2}}^{\frac{p}{2}-1}\left(-\frac{\lambda_{ij}^2}{2}\right)$ in the objective function of problem (P_1) complicates the optimization and makes the computation time too long. A way to simplify the optimization resolution is to calculate before the optimization a large set of Laguerre polynomial values corresponding to a large possible values of λ_{ij}^2 as the Laguerre polynomial presented in the expectation of the error Δ is related to the value of λ_{ij}^2. So we define Z a set in $\mathbb{R}$ which contains many possible values of λ_{ij}^2 that can be used during the optimization. For all $1 \leq i < j \leq n$, the value of λ_{ij}^2 is proportional to the distance $\|X_i + \theta_i - X_j - \theta_j\|_2^2$ thus the set Z is related to the data set by the value of d_{ij}^2 for all i, j as the objective of our optimization problem is to approach $\|X_i + \theta_i - X_j - \theta_j\|_2$ to d_{ij}. Therefore, we define Z as follows: $Z = [-\overline{d^2}/(4\sigma^2) + \ell, 0]$ where $\overline{d^2}$ is the mean of squared distances d_{ij} and ℓ is a value which gives the length of Z and related to the maximal value of d_{ij}^2 in order to cover the largest possible values of λ_{ij}^2. The increment between the elements of Z is taken equal to 10^{-2}. After that, during optimization, each value of λ_{ij}^2 calculated with a particular θ_i and θ_j is replaced by the nearest value in Z and the Laguerre polynomial value associated to this value is injected directly in the objective function. This simplification gives results close to the results obtained directly by optimizing (P_1) and reduce thousandth times the resolution time.

Moreover, the choice of σ and the scale parameter a are crucial to obtain good results. Therefore, the value of σ is taken equal to the mean of the p standard deviation calculated on the target matrix X, so we can write $\sigma = \dfrac{\sum_{k=1}^{p} \sigma_k}{p}$. Concerning the parameter a, we calculate it using the following algorithm:

Algorithm 2

Initialization: $\Theta^0 = (\theta_1^0 | \ldots | \theta_n^0) = (\mathbf{0} | \ldots | \mathbf{0})$.

for $t = 1$ to N_2 **do**

Solve problem (P_a) : $\min\limits_{a \in \mathbb{R}^+} \sum\limits_{1 \leq i < j \leq n} (d_{ij} - a\|X_i + \theta_i^{t-1} - X_j - \theta_j^{t-1}\|)^2$.

Solve problem (P_θ) : $\min\limits_{(\theta_1, \ldots, \theta_n) \in \mathbb{R}^p} \sum\limits_{1 \leq i < j \leq n} (d_{ij} - a^t\|X_i + \theta_i - X_j - \theta_j\|)^2$.

$\Theta^t = (\theta_1^t, \ldots, \theta_n^t)$ solution of problem (P_θ).

end for

NLopt library (Version 2.4.2) [11] implanted in language C a free/open-source library is used to solve problem (P_1) as this problem is a non-linear and non-convex optimization problem. In this library, numerous algorithms exist to solve such non-linear optimization problems. In our application, we choose Sbplx algorithm which is based on Subplex method [18] that is a generalization of Nelder-mead simplex.

Concerning simulation algorithm, the covariance matrix Σ is given by the covariance of matrix X multiply by a constant c. The parameter ξ presented in the proposal distribution is taken equal to 10^{-4}. Concerning the temperature parameter, we take it equal to $T = 100$. Moreover, during simulation, the number of iterations is taken equal to $N_1 = 300$.

3.7.3 Results

3.7.3.1 Optimization results

First, we want to define the different values of parameters a, σ and ℓ for the two data sets. Table 3.3 gives these values:

	a	σ	ℓ
White corn tortilla chips	26.37	0.55	1000
Muscadine grape juices	36.9	0.64	1000

Table 3.3: The values of parameters a, σ and ℓ for the two data sets.

The values of a for the two data sets are calculated using algorithm 2. Figure 3.2 depicts the trace plots of the value of a at each iteration for the two data sets. We show clearly that the value of a converge to an optimal value. Concerning the value of ℓ, a choice of 1000 for the two data set can be reasonable as the maximum value of the squared distances in the two data sets is in the range of 1000.

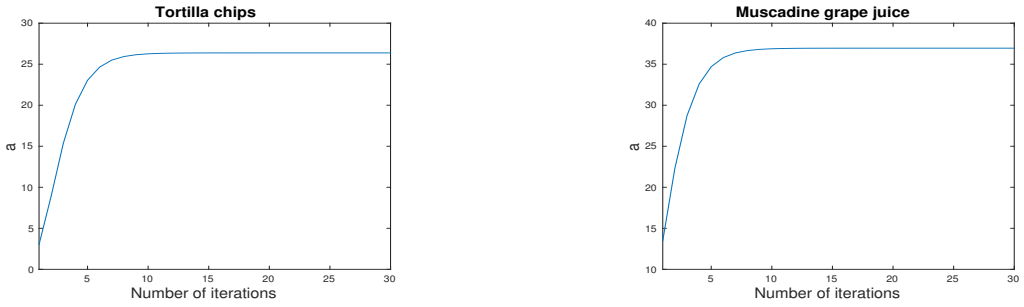


Figure 3.2: The trace plots of the results of the algorithm 2 for tortilla chips and muscadine grape juice data sets.

After parameters determination, the statistical test developed in Section 3.2.2 has been applied to perform the interest of the displacements of the points. The values of the ratio $\mathcal{R} = \dfrac{\mathbb{V}ar(\Delta)}{(\Delta_0 - \mathbb{E}(\Delta))^2}$ calculated for the two data sets are given in the Table 3.4. As the two values of $\mathcal{R}$ for the two data sets are smaller than 0.05, so the statistical test is significant for $\alpha = 0.05$. Therefore we reject the null hypothesis $(\mathcal{H}_0)$ and we accept the alternative hypothesis $(\mathcal{H}_1)$. Thus, the movements of points i and j through vectors θ_i and θ_j are

	$\mathcal{R}$
White corn tortilla chips	0.0239
Muscadine grape juices	0.0153

Table 3.4: The values of $\mathcal{R}$ for white corn tortilla chips and muscadine grape juices data sets.

necessary to approach the distances $\|X_i + \theta_i - X_j - \theta_j\|$ to d_{ij} for all $1 \leq i < j \leq n$. After the statistical test, problem (P_1) has been solved using different values of regularization parameter η.

Tables 3.5 and 3.6 show the different values of the expectation of error Δ and the number of non-null displacements θ_{ik} after optimization for different values of η obtained after optimization. We remark that when η increases, the number of displacements decreases and when η becomes too large, the number of displacements tends to zero and nothing moves.

White corn tortilla chips

η	$\mathbb{E}(\Delta)$	$\#(\theta_{ik} \neq 0)$
0	438896	407
10	438892	405
10^2	444855	401
10^3	393359	397
10^4	447001	360
10^5	709809	233
2×10^5	1450193	190
4×10^5	4558334	153
6×10^5	8229189	121
7×10^5	12330002	109
10^6	19927843	87
10^7	229240376	0

Table 3.5: The values of $\mathbb{E}(\Delta)$ and the number of non-null displacements for different values of η for tortilla chips data set.

Muscadine grape juices

η	$\mathbb{E}(\Delta)$	$\#(\theta_{ik} \neq 0)$
0	127845	150
10	127845	149
10^2	127845	147
10^3	127952	133
4×10^3	128799	128
6×10^3	130641	116
8×10^3	140656	98
10^4	168285	86
2×10^4	276341	54
4×10^4	551644	28
10^5	1336073	10
10^6	3517594	0

Table 3.6: The values of $\mathbb{E}(\Delta)$ and the number of non-null displacements for different values of η for muscadine juices data set.

A way to choose the value of η is to determine the number of misplaced points which must be moved to fit the distances. To find these misplaced points, we use the criterion of selection points ρ_i presented in Section 3.2.1.

Table 3.7 shows the values of this criterion. We have seen that for a fixed real number ϱ between 0 and 1, if $\rho_i > \varrho$ we consider i as misplaced point. So, by taking $\varrho = 0.1$ for the two data sets, we can detect 3 misplaced points for white corn tortilla chips and 4 for muscadine grape juices which is equivalent to $3 \times 37 = 111$ values of $\theta_{ik} \neq 0$ for tortilla chips and $4 \times 15 = 60$ for muscadine juices. Then, by referring to Tables 3.5 and 3.6, we choose the value of η that gives a number of displacement close to that obtained using the criterion ρ_i for tortilla chips and muscadine juices. Indeed, for tortilla chips data set, a value of η equal to 7×10^5 gives a number of displacements equal to 109 displacements that is close to 111. Similarly, we choose $\eta = 2 \times 10^4$ for muscadine juices data set. Noted that by changing the value of ϱ, we can detect more misplaced points so this choice must be reasonable.

i		1	2	3	4	5	6	7	8	9	10	11
ρ_i	$\mathcal{D}_1$	0.0640	**0.1253**	0.0706	0.0700	0.0931	0.0865	**0.1270**	0.0811	0.0828	**0.1088**	0.0908
	$\mathcal{D}_2$	0.0960	0.0589	0.0601	0.0900	0.0966	**0.1483**	0.0954	**0.1028**	**0.132**	**0.1197**	

Table 3.7: The values of criterion ρ_i for the 11 white corn tortilla chips samples ($\mathcal{D}_1$) and the 10 muscadine grape juices ($\mathcal{D}_2$). The bold values corresponds to the values where $\rho_i > 0.1$.

Besides, if the number of desirable displacements is fixed by the user then it is not needed to compute the ratio ρ and, in the same way, we can choose the value of η. So, the choice of η is always related to the objective which is aimed at.

The objective of the study is to determine the acceptable attributes categories of white corn tortilla chips and muscadine grape juices. Using our method we want to determine the product characteristics that must be changed to match with the consumers preference.

As we have seen we are interested in our method to fit the characteristics of product to the consumer acceptance rates, so null displacements can be interpreted as consumers satisfaction. Globally, for each categories of attributes, we can calculate the proportion of the null displacements. This proportion is given by:

$$p_{\mathrm{C}} = \frac{\text{number of } (\theta_{ik} = 0) \text{ in category } C_{\mathcal{D}}}{\text{total number of } \theta_{ik} \text{ in category } C_{\mathcal{D}}}$$

where $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2\}$, $\mathrm{C}_{\mathcal{D}_1} \in \{$Flavor, Texture, Appearance$\}$ is the category of white corn torilla chips ($\mathcal{D}_1$) and $\mathrm{C}_{\mathcal{D}_2} \in \{$ Basic tastes, Aromatics, Fellings factors $\}$ is the category of muscadine grape juices ($\mathcal{D}_2$). For each data set, the proportion p_{C} is calculated from Table 3.8 or Table 3.9.

				White	Corn	Tortilla	Chips				
	BYW	GMG	GUY	MED	MIS	MIT	OAK	SAN	TOB	TOM	TOR
					Flavor						
Sweet	0	0	0	1.2451	0	0	−0.9751	0	1.6118	1.3805	0
Salt	0	1.6930	0	1.9880	0	0	1.5206	0	0	0	0
sour	0	0	0	0	0	0	−2.7781	0.7170	0	0	0
Astringent	0	1.8238	0	0	−1.3989	0	0	0	0	−0.6097	1.8035
Grain complex	0	−1.3154	1.5591	2.3519	0	0	0	0	0	0	0
Raw corn	0	−1.9829	0	0	0	0	−2.8642	0	0	0	0
Masa	0	0	0	0	1.4716	0	−1.6456	0	0	−1.4004	0
Toasted grain	0	−1.2323	−1.3444	0	0	0	−0.4043	1.4624	1.0626	0	−1.5607
Heated oil	0	0	0	0	0	0	2.0665	0	0	−1.6899	0
Scorched	0	0	0	0	−3.1855	0	0	0	0	0	0
Cardboard	0.3165	1.3200	0	−1.9339	0	0	0	1.3330	1.6492	0	0
					Texture						
Oily/ greasy lip	−1.1890	1.4904	0	0	0	0	0	0	0	−1.5958	0
Loose particles	−1.8266	1.9488	0	0	0	0	0	0	0	0	0
Hardness	0	−2.3640	0	0	0	0	0	0	0	0	0
Crispness	0	1.3160	0	0	0	0	1.2906	0	−1.2140	0	−1.8826
Cohesiveness of mass	1.6169	−1.2832	0	0	0	0	−1.7938	0	0	0	−1.5653
Roughness of mass	−1.2758	1.3458	0	0	0	0	0	0	0	−1.8818	0
Moistness of mass	0	−1.2495	0	0	0	1.7023	−1.8192	0	0	0	0
Moisture absorption	0.9027	0	0	0	−1.4304	−1.7818	0	0	0	1.0449	0
Persistence of crisp	−2.2752	0	0	0	0	0	0.1331	0	−1.3031	0	0
Toothpack	0.1483	0	0	0	0	2.6819	0	0	−0.5108	0	0
					Appearance						
Degree of Whitenes	0	2.5915	0	0	0	1.9849	0	0	0	0	0
Grain Flecks	−0.8481	1.7684	0	0	0	1.7982	0	0	0	0	0
Char Marks	0	−1.1166	0	0	0	0	0	0	1.5467	0	1.5647
Micro Surface Particles	0	1.6609	0	0	0	0	0	0	0	−1.9241	0
Amount of Bubbles	0	0	0	0	0	0	0	0	−1.7983	0	1.0500

Table 3.8: The values of displacements θ_{ik} where $i = 1, \ldots, 11$ is the corn tortilla chip sample and k is the attributes of flavor, texture, appearance categories. Only the detected descriptive attributes are given in this table.

Tortilla attributes Categories	P_C
Flavor	0.71
Texture	0.72
Appearance	0.78

Muscadine attributes Categories	P_C
Basic tastes	0.70
Aromatics	0.60
Feeling factors	0.50

Table 3.10: The proportion values for different attributes categories for white corn tortilla chips and muscadine grape juices.

The proportion values for different categories for the two data sets are given in the Table 3.10. For white corn tortilla chips, the flavor, texture and appearance attributes categories have approximatively the same proportion values that it is equal in average to 0.75. So, 25% of the characteristics products must be moved to make the products characteristics as acceptable as possible by the consumers. Thus, we can conclude that the overall characteristics of tortilla chips are well accepted by the consumers. Concerning

	Muscadine grape juices									
	BB	CA	GV	IS	ME	CR	CW	SH	SUM	SUP
	Basic tastes									
Sweet	−1.1850	0.5018	0.6948	0	0	0	0	0	0	0
Sour	−1.0612	0.2443	0	0	0	−1.0543	0	0	0	0
	Aromatics									
Cooked muscadine	−0.1948	0	0	0	0	−0.4885	0.4100	0.5171	0	0
Cooked grape	0	−0.5705	0	0	0	−0.6065	0	0	−0.8350	0
Musty	0	−0.6460	0	−0.4153	0.7202	0	0.0497	0	0	0
Green/unripe	0	0.3416	0	0	0	0	0	−0.3443	0	0
Floral	0	0.4433	0.4345	0	−0.9344	0	−0.5002	0.4070	−0.9426	0
Apple/Pear	−1.0636	0	0.5652	0	0	−0.4278	0	0	0	0.6464
Fermented	−1.2588	0	0	0	0	−0.7521	0	0	0.3882	0
Metallic	0	0.6494	0	−1.1046	0.6358	0.6547	0	0.6520	−0.7677	0
	Feeling factors									
Astringent	0.3984	0	0	−0.8488	0	0.3228	0.7118	0	0	0.4081

Table 3.9: The values of displacements θ_{ik} where $i = 1, \ldots, 10$ is the 10 muscadine grape juices and k is the attributes of basic tastes, aromatics, feeling factors categories. Only the detected descriptive attributes are given in this table.

muscadine grape juices, we notice that the basic tastes attributes are the most acceptable attributes among the two other attributes categories as just 30% of the attributes must be changed to fit the consumer scores. Whereas, 40% and 50% of the aromatics and feeling factors attributes categories must respectively be changed to make these characteristics acceptable by the consumers.

3.7.3.2 Simulation results

Algorithm 1 has been applied to the two data sets. The constant c multiplied by covariance matrix of data is taken equal to 10^{-3} for the two data sets.

Figure 3.3 shows that the minimal value of error Δ obtained by simulation is equal to 18129 after 150 iterations for the white corn tortilla chips and 4838 after 50 iterations for the muscadine grape juices. The results of displacements for these two data sets are given in Figure 3.4 and 3.5. In these figures, we compare the displacements obtained by simulation with those obtained by optimization of problem P_0 (without penalization term) as the parsimonious choice of displacements is not taken into account in the simulation algorithm.

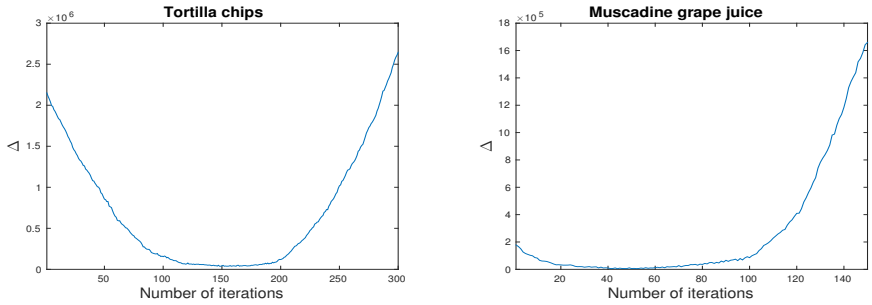


Figure 3.3: trace plot of the error Δ using algorithm 1 for white corn tortilla chips and muscadine grape juices data sets.

This comparison between optimization and simulation results indicates that using simulation technique we have succeeded in finding similar displacements with a value of Δ smaller than the value of the expectation calculated in the independent case. What is interesting here that the displacement obtained after simulation is not very different for most of the points. The important displacements obtained in the optimization and simulation results are close. So by taking some threshold to detect the important displacements, we can detect the same important displacements in two different ways.

3.7.4 Discussion

Several papers in food quality and preference domains study the relation between consumers preference and the characteristics of products in order to find the must acceptable characteristics of these products by the consumers [7]. Preference mapping techniques can be applied using just the consumers rates for each product, we address 'internal preference mapping', or by taking an additionally data describing the products with a series of criteria, we address 'external preference mapping' [3]. As our method based on the fitting of two matrices so the comparative with external preference mapping should be more explicative. The main objective to external preference mapping is to fit the individual consumer rates to the products configuration by using one of the different regression models among which the quadratic surface model is popular [4]. Meullenet in his paper [14] indicates that the preference mapping is determined by operating a partial least squares (PLS) regression model and the application of Jackknife optimization. This model is used to predict the consumer attributes acceptance. So, the prediction here is related to something subjective whereas our method gives displacements that can be interpretable without introducing subjective effects. Moreover, the displacement of attributes for each product can be interpretable alone or by taking all the categories of attributes. Thus, using our model we can find a new and simple methodology to determine the preference mapping of products.

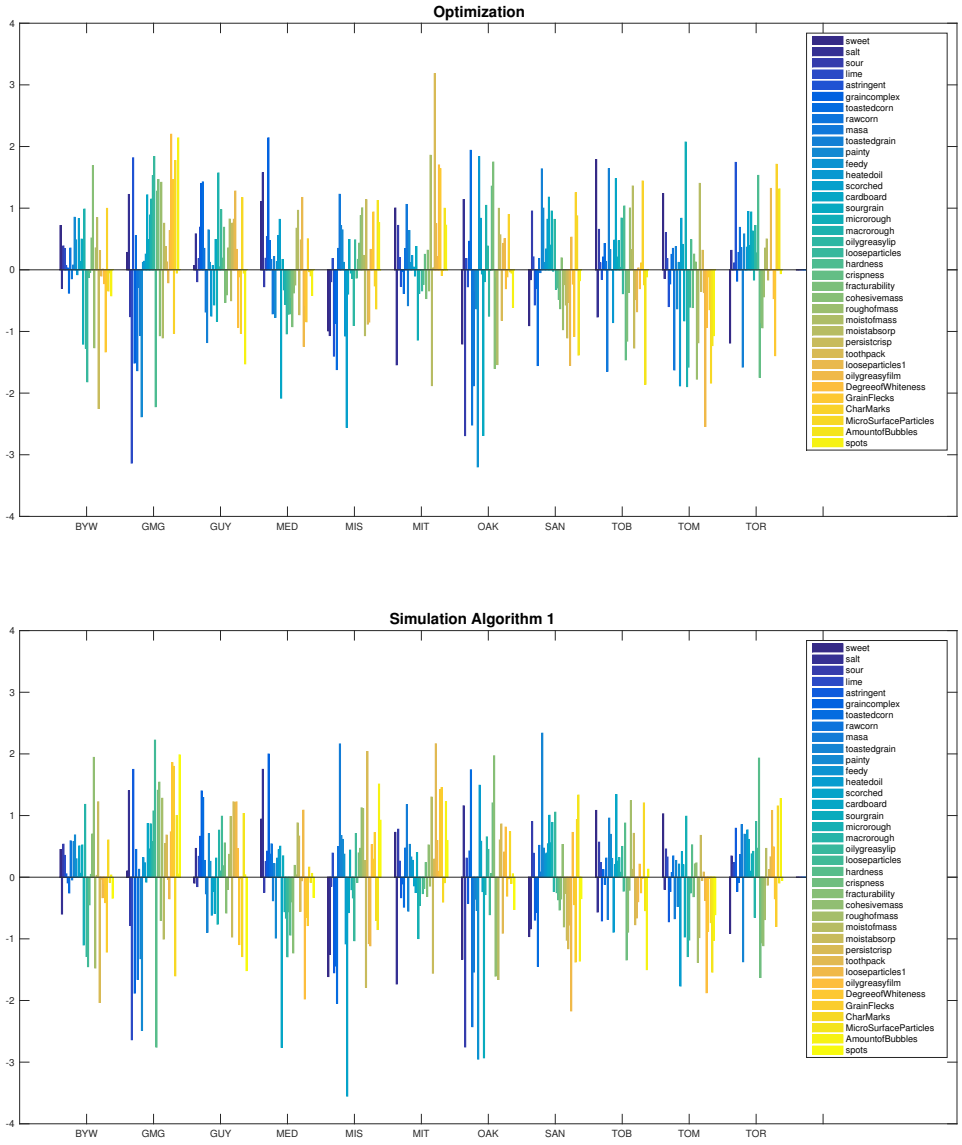


Figure 3.4: The displacements for the different attributes of the 11 tortilla chip samples obtained by optimization and simulation.

3.8 Conclusion

We have presented a new model of multidimensional fitting method by taking into account random effects. First, the random model of MDF with a penalized form is presented. Second, a statistical test indicates the significance of the displacements of the points. Then, optimization and simulation algorithms are developed to find these displacements. The application of this method in the sensometrics domain shows the simplest explanation of the sensory profiles of products according the consumers preference. Finally, MDF in their deterministic and random model can be also used when the data contains missing

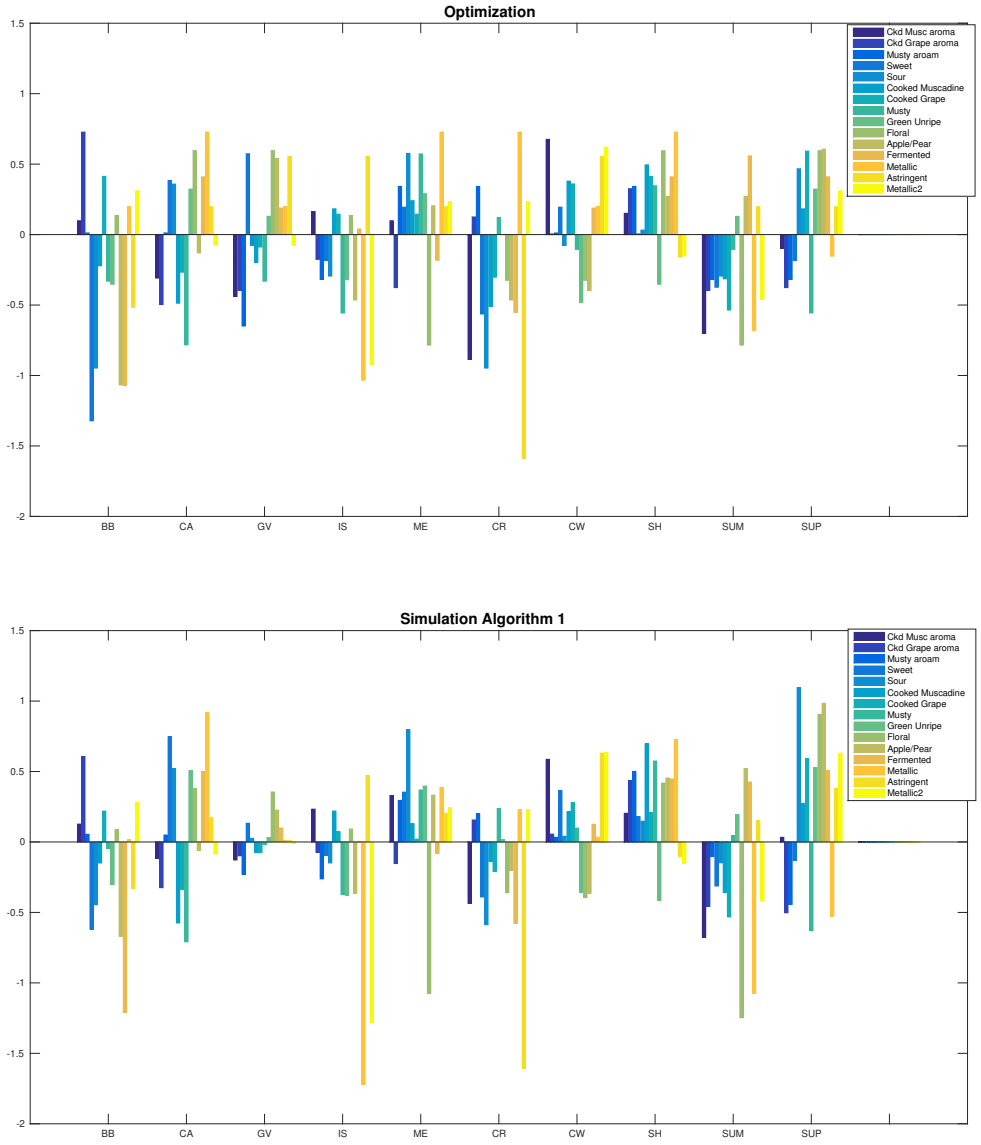


Figure 3.5: The displacements for the different attributes of the 10 muscadine grape juices obtained using optimization and simulation.

data. A pretreatment of this data before the application of MDF method to replace these missing values will not impinge the results.

Bibliography

[1] Akaike, H. Information theory and an extension of the maximum likelihood principle. In Proc. 2nd Int. Symp. Information Theory (eds B. N. Petrov and F. Csaki). Budapest: Akademiai Kiado, 1973.

[2] Allen, D.M. The relationship between variable selection and data augmentation and a method for prediction. Technometrics, 16, 125-127, 1974.

[3] Carroll, J. D. Individual differences and multidimensional scaling. In R. N. Shepard, A. K. Romney, and S. Nerlove (Eds.)Multidimensional Scaling: Theory and Applications in the Behavioral Sciences. 105–155, New York:Seminar Press, 1972.

[4] Danzart, M. Quadratic model in preference mapping. 4th Sensometric meeting, Copenhagen, Denmark, 1998.

[5] Filaseta, M. and Lam, T.Y., On the irreducibility of the generalized Laguerre polynomials. Acat Arithmetica, 105(2), 177–182, 2002.

[6] Golub , H.G., Heath, M. and Wahba , G. Generalized Cross-Validation as a Method for Choosing a Good Ridge Parameter. Technometrics, 21(2), 215–223, 1979.

[7] Greenhoff, K. and MacFie, H. J. H., Preference mapping in practice. In H. J. H. MacFie and D. M. H. Thomson (Eds.), Measurement of Food Preferences, 137–165, Springer US, Boston, MA, 1994.

[8] Green, P.J., Penalized Likelihood for General Semi-Parametric Regression Models. international Statistical Review Revue Internationale de Statistique , Vol. 55, No. 3, 245–259, 1987.

[9] Hastings, W. Monte Carlo sampling methods using Markov chains and their application. Biometrika, 57, 97–109, 1970.

[10] Hoerl, A.E. and Kennard, R. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12, 55–67, 1970.

[11] Johnson, S.G. The NLopt nonlinear-optimization package, http://ab-initio.mit.edu/nlopt, 2008.

[12] Lawless, H. T., Heymann, H. Sensory Evaluation of Food: Principles and Practices. Gaithersburg, Maryland: Aspen Publishers, Inc, 1999.

[13] Meilgaard, M. C., Civille, G. V., Carr, B. T. Sensory Evaluation Techniques. Boca Raton, Florida: CRC Press, 2007

[14] Meullenet, J.F., Xiong, R., Monsoor, M.A., Bellman-Homer, T., Dias, P., Zivanovic, S., Fromm, F. and Liu, Z. Preference Mapping of Commercial Toasted White Corn Tortilla Chips. Journal of Food Science, 67(5), 1950–1957, 2002.

[15] Meullenet, J.-F., Xiong, R., Findlay, C.J. Multivariate and Probabilistic Analyses of Sensory Science Problems. Ames, IA, USA: Blackwell Press, 2007.

[16] Meullenet, J.F., Lovely, C., Threlfall, R., Morris, J.R., Striegler, R.K. An ideal point density plot method for determining an optimal sensory profile for Muscadine grape juice. Food Quality and Preference, 19, 210–219, 2008.

[17] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. Equations of state calculations by fast computing machines. J. Chem. Phys., 21(6), 1087–1092, 1953.

[18] Rowan, T. Functional Stability Analysis of Numerical Algorithms. Ph.D. thesis, Department of Computer Sciences, University of Texas at Austin, 1990.

[19] Schwarz, G. Estimating the dimension of a model. Ann. Statist.,6, 461–464, 1978.

[20] Stone, M. Cross-validation choice and assessment of statistical predictions. Journal of the Royal Statistical Society: Series B 36, 111-147, 1974.

Chapter 4

Projection under pairwise distance control

In this chapter, we have developed a new dimensionality reduction method defined as a non-linear projection method that takes into account the projection quality of each projected point in the reduced space, this quality being directly available in the same scale as this reduced space. More specifically, this novel method allows a straightforward visualization data in $\mathbb{R}^2$ with a simple reading of the approximation quality, and provides then a novel variant of dimensionality reduction.

4.1 Introduction

In chapter 1, it was seen that a large data dimensionality reduction and data visualization methods have been proposed to drop the difficulties associated to the high dimensional data. Principal component analysis, multidimensional scaling (MDS), scatter plot matrix, parallel coordinates and Sammon's mapping are some of the known used methods.
Scatter plot matrix, parallel coordinates and Sammon's mapping methods are widely used to visualize multidimensional data sets. The first two methods have as inconvenient that when the number of dimensions grows, important dimensional relationships might not be visualized. Concerning Sammon's mapping method, the inconvenient is similar to that found in PCA and MDS from the point of view of projection quality. Indeed, the quality of projection assessed by the percentage of variance that is conserved or by the stress factor is a global quality measure and takes only into account what happens globally but in some projection methods like PCA, a local measure is defined to indicate the projection quality of each projected point taken individually. This local measure is evaluated by the squared cosine of angle between the principal space and the vector of the point. A good representation in the projected space is hinted by high squared cosine values. This measure is useful in cases of linear projection as happens in PCA but cannot be applied

to the case of non-linear projection.

In this chapter, we propose a new non-linear projection method that projects the points in a reduced space by using the pairwise distance between pairs of points and by taking into account the projection quality of each point taken individually. This projection leads to a representation of the points as circles with a different radius associated to each point. Henceforth, this method will be called "Projection under pairwise distance control". The main contributions of this study are to give a simple data visualization in $\mathbb{R}^2$ with a straightforward interpretation and provide a new variant of dimensionality reduction. First, the new projection method is presented in Section 2. Then, in Section 3, the algorithms used in the resolution of optimization problems related to this method are illustrated. Next, Section 4 shows the application of this method to various real data sets. Finally, Section 5 concludes this work.

4.2 Projection under pairwise distance control

Let us consider n points given by their pairwise distance noted d_{ij} for $i, j \in \{1, \ldots, n\}$. The task here is to project these points using distances into a reduced space $\mathbb{R}^k$ by introducing additional variables, called hereafter radii, that indicate to which extent the projection of each point is accurate. The local quality is then given by the values of the radii. A good quality projection of point i is indicated by a small radius value noted r_i. It will be important to note that both units of d_{ij}'s and r_i's are identical, allowing direct comparison.

Before developing our method, an overview of principal component analysis (PCA) is presented to highlight the interest of our method.

4.2.1 Principal Component Analysis (PCA)

PCA method is the most used method in the data visualization and dimensionality reduction. This method is a linear projection technique applied when the data is linearly separable. PCA problem can be stated as an optimization problem involving the squared Euclidean distances [14]. This optimization problem is the following:

$$\mathcal{P}_{\text{PCA}} : \begin{cases} \min\limits_{A \in \mathcal{M}_{p \times q}} \sum\limits_{1 \leq i < j \leq n} |d_{ij}^2 - \|Ay_i - Ay_j\|^2| \\ s.t. \;\; \text{rank}(A) = k \\ \qquad AA^T = I_p \end{cases}$$

where $y_i \in \mathbb{R}^p$ is the original coordinates vector of point i, d_{ij}^2 is the squared distance for couple (i, j) given by $\|y_i - y_j\|^2$ and A is the projection matrix of dimension $q \times p$ with q is the reduced space dimension.

By construction, PCA cannot take into account non-linear structures, since it describes the data in terms of a linear subspace. Furthermore, the only measures used to evaluate

the projection quality of points are the squared cosines values which can only be used in the case of linear projection. Thus, the individual control of projection is no more guaranteed using non-linear projection method.
Moreover if we consider now n variables $r_1, \ldots, r_n \in \mathbb{R}^+$, the sum of which bounds the objective function, the PCA optimization problem $\mathcal{P}_{\text{PCA}}$ can be equivalently rewritten as:

$$\mathcal{P}_{\text{PCA}} : \begin{cases} \min\limits_{r_1,\ldots,r_n \in \mathbb{R}^+, A \in \mathcal{M}_{p\times q}} \sum\limits_{i=1}^{n} r_i \\ s.t. \quad \sum\limits_{i=1}^{n} r_i \geq \dfrac{1}{n-1} \sum\limits_{1\leq i<j\leq n} |d_{ij}^2 - \|Ay_i - Ay_j\|^2| \\ \quad \text{rank}(A) = k \\ \quad AA^T = I_p \end{cases}$$

Then, we can observe that the constraint on $\sum_{i=1}^{n} r_i$ can be modified to have a stronger control on each d_{ij} in the following way: $|d_{ij} - \|x_i - x_j\|| \leq r_i + r_j$ where x_i and x_j are projection coordinates of points i and j. Here the projection coordinates are not obtained necessarily by linear projection anymore.

Our new non-linear projection method that controls individually the projection of points is developed hereafter.

4.2.2 Our proposed method

Let $x_1, \ldots, x_n$ be the coordinates of the projected points in $\mathbb{R}^k$. Radii are an important element of the paper introduced to assess how much the distance between two projected points (i, j) given by $\|x_i - x_j\|$ is far from given distance d_{ij}. Indeed, radii (r_i, r_j) for couple (i, j) are small when $\|x_i - x_j\|$ is close to d_{ij}. Figure 4.1 depicts this idea: for all points $i, j \in \{1, \ldots, n\}$ the projected point of each point i belongs to a sphere with center x_i and radius r_i such that $\|x_i - x_j\| - r_i - r_j \leq d_{ij} \leq \|x_i - x_j\| + r_i + r_j$.

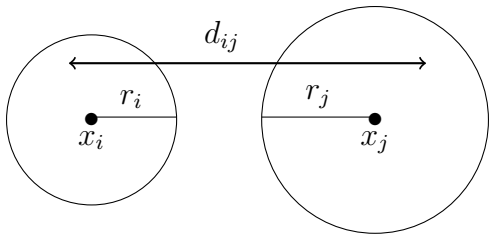


Figure 4.1: Examples of radii for bounding the original distance d_{ij}

This idea can be expressed by finding the value of radii that satisfy these two constraints:

- $\sum_{i=1}^{n} r_i$ is minimum.

- $d_{ij} \in \{\|x_i - x_j\| - r_i - r_j, \|x_i - x_j\| + r_i + r_j\}$, for $1 \leq i < j \leq n$.

The projection under pairwise distance control problem can be written as the following optimization problem:

$$\mathcal{P}_{r,x} : \begin{cases} \min\limits_{r_1,\ldots,r_n \in \mathbb{R}^+, x_1,\ldots,x_n \in \mathbb{R}^k} \sum\limits_{i=1}^{n} r_i \\ s.t \quad |d_{ij} - \|x_i - x_j\|| \leq r_i + r_j, \text{ for } 1 \leq i < j \leq n \end{cases}$$

Of course, by fixing the coordinates vectors x_i for all $i \in \{1, \ldots, n\}$, using principal component analysis or any other projection method, the problem can easily be solved in $(r_1, \ldots, r_n)$ using linear programming. This problem can be written as follows:

$$\mathcal{P}_{r} : \begin{cases} \min\limits_{r_1,\ldots,r_n \in \mathbb{R}^+} \sum\limits_{i=1}^{n} r_i \\ s.t \quad |d_{ij} - \|x_i - x_j\|| \leq r_i + r_j, \text{ for } 1 \leq i < j \leq n \end{cases}$$

We can remark that a solution of $\mathcal{P}_r$ always exists. Indeed, to satisfy the constraints it is enough to increase all r_i. Besides, solving $\mathcal{P}_r$ with fixed coordinates $(x_1, \ldots, x_n)$ does not lead in general to the optimum of problem $\mathcal{P}_{r,x}$.

4.2.3 Visualization example

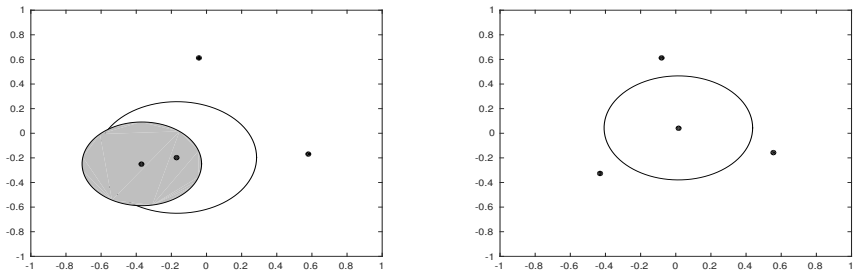
Let us apply our projection method to a simple example by taking a tetrahedron with all pairwise distance equal to 1. For problem $\mathcal{P}_r$, the coordinates x_i for $i = 1, \ldots, 4$ are obtained using multidimensional scaling. Using linear and non-linear optimization packages in Matlab respectively for problems $\mathcal{P}_r$ and $\mathcal{P}_{r,x}$ give a value of $\sum_{i=1}^{n} r_i$ equal to 0.7935 for problem $\mathcal{P}_r$ and 0.4226 for $\mathcal{P}_{r,x}$. Figure 4.2a corresponds to the first solution and Figure 4.2b corresponds to the second one. In Figures 4.2*a* and 4.2*b*, we depict circles with different radii. The circle color is related to the radius values, the shades of gray lie between white and black in the descending direction of the radius values. The smaller the radius, the darker circle. The points that have circles with small radii are considered as well projected points. Note that the points that are represented as points and not circles are very well projected, having radii almost equal to zero. In Figure 4.2*a*, half of the points is well projected whereas the other half have large radii indicating that they are not well projected. In Figure 4.2*b* just one circle appears marking that the projection quality using problem $\mathcal{P}_{r,x}$ is better than $\mathcal{P}_r$.

4.2.4 Link with other methods

We have seen Multidimensional fitting (MDF) is a method that modifies the coordinates of a set of points in order to make the distances calculated on the modified coordinates similar to given distances on the same set of points.

The objective function of MDF problem is given by:

$$\sum_{1 \leq i < j \leq n} |d_{ij} - \|x_i - x_j\||.$$



(a) $\sum_{i=1}^{n} r_i = 0.7935$ (b) $\sum_{i=1}^{n} r_i = 0.4226$

Figure 4.2: Projected points after solving $\mathcal{P}_r$ and $\mathcal{P}_{r,x}$. (a) shows the projection obtained from the solution of $\mathcal{P}_r$ using MDS and (b) shows that obtained from the solution of $\mathcal{P}_{r,x}$.

Property 4.1. *Problem $\mathcal{P}_{r,x}$ is bounded below by $\frac{1}{n-1} \sum_{1 \leq i < j \leq n} |d_{ij} - \|x_i - x_j\||$ where $x_1, \ldots, x_n$ is the optimum of the associated MDF problem.*

Proof. By summing all the constraints of problem $\mathcal{P}_{r,x}$ we obtain:

$$\sum_{1 \leq i < j \leq n} |d_{ij} - \|x_i - x_j\|| \leq \sum_{1 \leq i < j \leq n} r_i + r_j = (n-1) \sum_{i=1}^{n} r_i$$

So, $\sum_{i=1}^{n} r_i \geq \frac{1}{n-1} \sum_{1 \leq < i < j \leq n} |d_{ij} - \|x_i - x_j\||$, which concludes the proof. □

4.3 Lower Bound of the optimization problem of the projection under pairwise distance control method

Minimization problem $\mathcal{P}_{r,x}$ is too hard to be solved exactly. A way to assess how good a solution is, is to provide a lower bound on the objective function. Then, if the bound is close to the best found solution, we can conclude that this solution is fixed. Thus, in this section we present such a lower bound of problem $\mathcal{P}_{r,x}$.
Let $x_1, \cdots, x_n; r_1, \cdots, r_n$ a feasible solution of $\mathcal{P}_{r,x}$ and $M \in \mathbb{R}$ such that $M = \max_{(i,j)} \| x_i - x_j \|$.
We consider three functions noted f, g, h depending on M as follows:

- $f(M) = \sqrt{\left(1 - \frac{n}{3}\right) M^2 + \frac{1}{n-1} \sum_{i<j} d_{ij}^2} - M.$
- $g(M) = |M - d_{max}|.$
- $h(M) = \min_{i<j} \max_{k<l; k,l \neq i,j} \min \left\{ L^1_{ijkl}; L^2_{ijkl} \right\}.$

where:

- $d_{max} = \max\limits_{1\leq i<j\leq n}\{d_{ij}\}$,
- $L^1_{ijkl} = \max\left\{\dfrac{d_{jk}-d_{kl}}{2}; \dfrac{d_{jl}-d_{kl}}{2}; |d_{ij}-M|\right\}$,
- $L^2_{ijkl} = \max\left\{\dfrac{d_{ik}-d_{kl}}{2}; \dfrac{d_{il}-d_{kl}}{2}; |d_{ij}-M|\right\}$.

Using results presented in section 4.3.1.2, we can write:

$$\begin{cases} \sum\limits_{i=1}^{n} r_i \geq f(M) \\ \sum\limits_{i=1}^{n} r_i \geq g(M) \\ \sum\limits_{i=1}^{n} r_i \geq h(M)\cdot \end{cases}$$

Consequently,

$$\sum_{i=1}^{n} r_i \geq \max\{f(M); g(M); h(M)\}\cdot \tag{4.1}$$

The inequality (4.1) is true for all solutions of $\mathcal{P}_{r,x}$ particularly for the optimal solution. Thus:

$$\sum_{i=1}^{n} r_i^{opt} \geq \max\{f(M); g(M); h(M)\}\cdot \tag{4.2}$$

Hence, the lower bound is given by:

$$\sum_{i=1}^{n} r_i^{opt} \geq \min_{M} \max\{f(M); g(M); h(M)\} \text{ for all feasible solutions}\cdot$$

Given M, a lower bound of problem $\mathcal{P}_{r,x}$ is derived. Afterwards, a bound free of M is given by minimizing the bounds depending on M.

4.3.1 Construction of function f, g and h

Three function are used to find a lower bound of our objective function. The construction of function f requires the use of the result presented in Lemma 4.2 cited bellow. Therefore, we have presented first of all some results shown as lemmas and corollary and then we have defined the three functions.

4.3.1.1 Two Lemmas used

Let (C) be a circle with center O and radius r. Let consider n points with coordinates $x_1, \cdots, x_n$ such that for all $i = 1, \ldots, n$, $\|x_i - O\| \leq r$ and having g as center of gravity. This hypothesis is used in Lemma 4.1 and Corollary 4.1.

Lemma 4.1. *For all points $x_1, \ldots, x_n$, we have:*

$$\|x_i - O\| = r \text{ when } \sum_{i=1}^{n} \|x_i - g\|^2 \text{ is maximum .}$$

Proof. Let A and B two points among the n points such that A is inside the circle (C) and B belonging (C). The point A and B have as coordinates (a_1, a_2) and $(r, 0)$ respectively. We want to show that by moving B by small movements along the circle, we can approach the point A to the circle border increasing thus the inertia $\sum_{i=1}^{n} \|x_i - g\|^2$.
We note B', A' the new positions after movements of B and A respectively. Let θ be the angle between (OB) and (OB'), u_θ the displacement of B and (a'_1, a'_2) the coordinate of point A'.
Approaching A to the circle border requires the opposite movements of A and B with equal length. This constraint is necessary to keep the center in the same position. We distinguish two cases:

1- A having $a_2 < 0$.

2- A having $a_2 > 0$.

The two cases are illustrated in Figure 4.3.

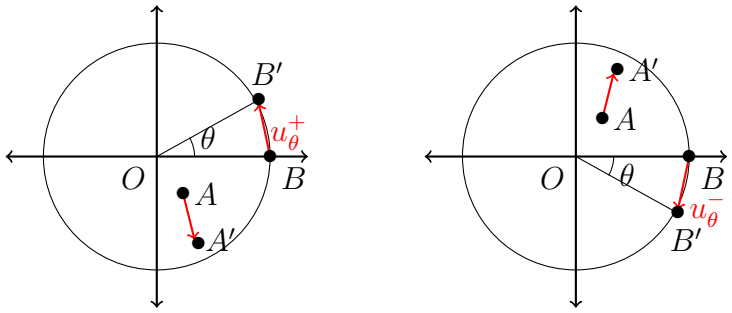


Figure 4.3: Representation of movements of points A and B in cases 1 and 2.

For case 1, A in the lower half of the circle requires that B moves positively with angle $\theta \in [0; \frac{\pi}{2}]$. In this case, the vector u_θ is given by: $u_\theta^+ = \begin{pmatrix} \cos\theta - 1 \\ \sin\theta \end{pmatrix}$ and the inner product $\langle u_\theta^+, AA' \rangle$ is given by:

$$\langle u_\theta^+, AA' \rangle = (a'_1 - a_1)(\cos\theta - 1) + (a'_2 - a_2)\sin\theta \cdot \tag{4.3}$$

Here, we have $a'_1 \geq a_1$ and $a'_2 \leq a_2$ that imply $a'_1 - a_1 \geq 0$ and $a'_2 - a_2 \leq 0$. Moreover, we have $0 \leq \cos\theta \leq 1$ and $\sin\theta \geq 0$ that give $\langle u_\theta^+, AA' \rangle < 0$.

For case 2, A in the upper half of the circle requires that B moves negatively with angle $\theta \in [3\frac{\pi}{2}, 2\pi]$. In this case, the vector u_θ is given by: $u_\theta^- = \begin{pmatrix} \cos\theta - 1 \\ -\sin\theta \end{pmatrix}$ and the inner product is given by:

$$\langle u_\theta^-, AA' \rangle = (a'_1 - a_1)(\cos\theta - 1) - (a'_2 - a_2)\sin\theta \tag{4.4}$$

Here, we have $a'_1 \geq a_1$ and $a'_2 \leq a_2$ that imply $a'_1 - a_1 \geq 0$ and $a'_2 - a_2 \leq 0$. Moreover, we have $0 \leq \cos\theta \leq 1$ and $\sin\theta \leq 0$ that give $\langle u_\theta^-, AA' \rangle < 0$. □

Corollary 4.1. *The center of gravity g of $x_1, \ldots, x_n$ is the center of circle (C) i.e. $O = g$ when $\sum_{i=1}^{n} \|x_i - g\|^2$ is maximum.*

Proof. We have:

$$\begin{aligned}
\sum_{i=1}^{n} \|x_i - g\|^2 &= \sum_{i=1}^{n} \|x_i - O + O - g\|^2 \\
&= \sum_{i=1}^{n} \|x_i - O\|^2 + \sum_{i=1}^{n} \|O - g\|^2 + 2\sum_{i=1}^{n} (x_i - O)'(O - g) \\
&= \sum_{i=1}^{n} \|x_i - O\|^2 + \sum_{i=1}^{n} \|O - g\|^2 + 2n(g - O)'(O - g) \\
&= \sum_{i=1}^{n} \|x_i - O\|^2 + n\|O - g\|^2 - 2n\|O - g\|^2 \\
&= \sum_{i=1}^{n} \|x_i - O\|^2 - n\|O - g\|^2
\end{aligned}$$

All the points belong to the circle (C) as a result of Lemma 4.1. So, $\|x_i - O\|^2$ is fix and equal to r^2. Thus, maximizing inertia $\sum_{i=1}^{n} \|x_i - g\|^2$ amounts to minimize $\|O - g\|^2$. Then, the minimum of $\|O - g\|^2$ is zero so that $O = g$. □

Lemma 4.2. *If $\sum_{i=1}^{n} \| x_i - g \|^2$ is maximum for points $x_1, \ldots, x_n$ under constraints, for all couple (i, j), $\|x_i - x_j\| \leq M$, then an upper bound of $\sum_{i=1}^{n} \| x_i - g \|^2$ is given by: $\frac{nM^2}{3}$.*

Proof. Let (C) be the smallest circle containing the n points with coordinates $x_1, \ldots, x_n$. We consider three points noted A, B and C among the n points and having x_A, x_B and x_C as coordinates . We suppose that A, B and C belong to the circle (C) and the distance between B and C is equal to M *i.e.* $\|x_B - x_C\| = M$. By hypothesis, we have $\|x_A - x_B\| \leq M$. We note θ the angle between (AB) and (BC). Figure 4.4 illustrates the situation.
If $\theta > \frac{\pi}{3}$ then $\|x_A - x_B\| \leq M$ and $\|x_A - x_C\| > M$. By reversing the role of x_B and x_C we get $\theta = \frac{\pi}{3}$. So, $A = A'$ (A' is at the position indicated in figure 4.4) and then $r = \frac{M}{\sqrt{3}}$. Now, if we consider n points $y_1, \ldots, y_n$ and using Lemma 4.1 and Corollary 4.1, the maximum of the inertia $\sum_{i=1}^{n} \| y_i - g \|^2$, under the constraint that $y_1, \ldots, y_n$ are inside (C) is equal to $nr^2 = n\frac{M^2}{3}$. Thus, the maximum of $\sum_{i=1}^{n} \| x_i - g \|^2$ is upper bounded by $n\frac{M^2}{3}$. □

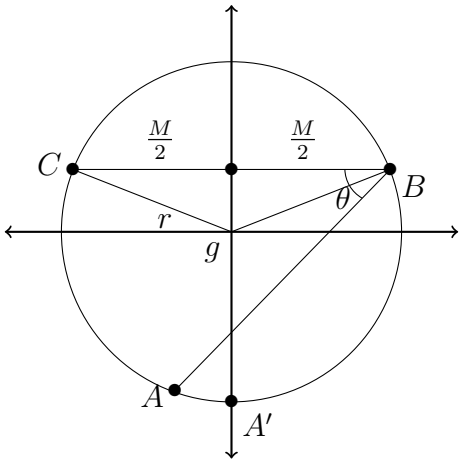


Figure 4.4: Representation of the points on the circle.

4.3.1.2 The three functions

Function $f(M)$ Using the constraints of problem $\mathcal{P}_{r,x}$, we have:

$$\begin{aligned} d_{ij} &\leq \| x_i - x_j \| + r_i + r_j \\ d_{ij}^2 &\leq (\| x_i - x_j \| + r_i + r_j)^2 \\ \sum_{i<j} d_{ij}^2 &\leq \sum_{i<j} \| x_i - x_j \|^2 + \sum_{i<j} (r_i + r_j)^2 + 2 \sum_{i<j} (\| x_i - x_j \|)(r_i + r_j) \\ \sum_{i<j} d_{ij}^2 &\leq \sum_{i<j} \| x_i - x_j \|^2 + \sum_{i<j} (r_i + r_j)^2 + 2M \sum_{i<j} (r_i + r_j) \text{ as } \| x_i - x_j \| \leq M \end{aligned} \quad (4.5)$$

Let g be the center of gravity of the projected points $x_1, \ldots, x_n$, so:

$$\begin{aligned} \| x_i - x_j \| &= \| x_i - g - x_j + g \| \\ \| x_i - x_j \|^2 &= \| x_i - g \|^2 + \| x_j - g \|^2 + 2(x_i - g)'(x_j - g) \\ \sum_{i<j} \| x_i - x_j \|^2 &= \sum_{i<j} \left(\| x_i - g \|^2 + \| x_j - g \|^2 \right) + 2 \sum_{i<j} (x_i - g)'(x_j - g) \cdot \end{aligned}$$

As $\sum_{i<j} (x_i - g)'(x_j - g) = 0$ then:

$$\sum_{i<j} \| x_i - x_j \|^2 = (n-1) \sum_{i=1}^{n} \| x_i - g \|^2 \cdot \quad (4.6)$$

Replacing equation (4.6) in (4.5) gives :

$$(n-1) \sum_{i=1}^{n} \| x_i - g \|^2 + \sum_{i<j} (r_i + r_j)^2 + 2M \sum_{i<j} (r_i + r_j) - \sum_{i<j} d_{ij}^2 \geq 0 \cdot \quad (4.7)$$

The quantity $\sum_{i=1}^{n} \| x_i - g \|^2$ is the inertia of the projected points $x_1, \ldots, x_n$. As long as we want to conserve the initial information, the inertia must be maximal under constraints $\|x_i - x_j\| \leq M$ for all $1 \leq i < j \leq n$.

Recalling equation (4.7) and by using Lemma 4.2, we obtain:

$$\frac{n(n-1)}{3}M^2 + \sum_{i<j}(r_i + r_j)^2 + 2M\sum_{i<j}(r_i + r_j) - \sum_{i<j} d_{ij}^2 \geq 0$$

$$\frac{n(n-1)}{3}M^2 + (n-1)\left(\sum_{i=1}^{n} r_i\right)^2 + 2(n-1)M\sum_{i=1}^{n} r_i - \sum_{i<j} d_{ij}^2 \geq 0$$

$$\left(\sum_{i=1}^{n} r_i\right)^2 + 2M\left(\sum_{i=1}^{n} r_i\right) + \frac{n}{3}M^2 - \frac{1}{n-1}\sum_{i<j} d_{ij}^2 \geq 0. \tag{4.8}$$

The discriminant of equation (4.8) is given by: $\Delta = 4\left(1 - \frac{n}{3}\right)M^2 + \frac{4}{n-1}\sum_{i<j} d_{ij}^2$ and as $r_i \geq 0$, $\forall i = 1, \ldots, n$ we get:

$$\sum_{i=1}^{n} r_i \geq \sqrt{\left(1 - \frac{n}{3}\right)M^2 + \frac{1}{n-1}\sum_{i<j} d_{ij}^2} - M.$$

We note $f(M) = \sqrt{\left(1 - \frac{n}{3}\right)M^2 + \frac{1}{n-1}\sum_{i<j} d_{ij}^2} - M.$

Function $g(M)$ Two situations are possible:

1. $\exists(i', j')$ such that $\| x_{i'} - x_{j'} \| = M$, that gives:
$$r_{i'} + r_{j'} \geq \| x_{i'} - x_{j'} \| - d_{i'j'} \geq M - d_{i'j'} \geq M - d_{max}.$$
As $\sum_{i=1}^{n} r_i \geq r_{i'} + r_{j'}$, we obtain:
$$\sum_{i=1}^{n} r_i \geq M - d_{max}$$

2. $\exists(i^*, j^*)$ such that $d_{i^*j^*} = d_{max}$, that gives:
$$r_{i^*} + r_{j^*} \geq d_{i^*j^*} - \| x_{i^*} - x_{j^*} \| \geq d_{max} - M.$$
Then, we obtain:
$$\sum_{i=1}^{n} r_i \geq d_{max} - M.$$

Hence:

$$\sum_{i=1}^{n} r_i \geq |M - d_{max}|. \tag{4.9}$$

We note $g(M) = |M - d_{max}|$.

Function $h(M)$ Let us consider four distinct points i, j, k and l. We suppose that there is a couple (i, j) such that $\|x_i - x_j\| = M$ and one of their coordinates is equal to zero ($x_i = 0$ or $x_j = 0$). We distinguish two cases:

1. $x_i = 0$.

2. $x_j = 0$.

Case 1: For $x_i = 0$, we take $x_j = \alpha x_k + \beta x_l$ with $\alpha, \beta \in [0,1]$. The constraints related to these four points are the following:

$$\begin{cases} |\|x_j\| - d_{ij}| \leq r_i + r_j & \text{(C1)} \\ |\|x_k\| - d_{ik}| \leq r_i + r_k & \text{(C2)} \\ |\|x_l\| - d_{il}| \leq r_i + r_l & \text{(C3)} \\ |\|x_j - x_k\| - d_{jk}| \leq r_j + r_k & \text{(C4)} \\ |\|x_j - x_l\| - d_{jl}| \leq r_j + r_l & \text{(C5)} \\ |\|x_k - x_l\| - d_{kl}| \leq r_k + r_l & \text{(C6)} \end{cases}$$

Firstly, using constraints (C4) and (C6) we obtain:

$$2\sum_{t=1}^{n} r_t \geq d_{jk} - d_{kl} + \|x_k - x_l\| - \|x_j - x_k\|\cdot$$

Additionally, as $x_j = \alpha x_k + \beta x_l$ with $\alpha, \beta \in [0,1]$, then

$$\|x_k - x_j\| = \|x_k - \alpha x_k - \beta x_l\| = \|(1-\alpha)x_k - \beta x_l\| \leq \|x_k - x_l\|,$$

which gives

$$\sum_{t=1}^{n} r_t \geq \frac{d_{jk} - d_{kl}}{2}\cdot \tag{4.10}$$

Secondly, using constraints (C5) and (C6) we obtain:

$$2\sum_{t=1}^{n} r_t \geq d_{jl} - d_{kl} + \|x_k - x_l\| - \|x_j - x_l\|\cdot$$

As $\|x_l - x_j\| \leq \|x_k - x_l\|$ we obtain:

$$\sum_{t=1}^{n} r_t \geq \frac{d_{jl} - d_{kl}}{2}\cdot \tag{4.11}$$

Thirdly, constraint (C1) and initial hypothesis $\|x_i - x_j\| = M$ lead to:

$$\sum_{t=1}^{n} r_t \geq |d_{ij} - M|\cdot \tag{4.12}$$

Consequently, equations (4.10), (4.11) and (4.12) involve:

$$\sum_{t=1}^{n} r_t \geq \max\left\{\frac{d_{jk} - d_{kl}}{2}; \frac{d_{jl} - d_{kl}}{2}; |d_{ij} - M|\right\} \text{ denoted } L^1_{ijkl}$$

Case 2: For $x_j = 0$, we take $x_i = \alpha x_k + \beta x_l$ with $\alpha, \beta \in [0,1]$. By analogy with case 1, we obtain:

$$\sum_{t=1}^{n} r_t \geq \max\left\{\frac{d_{ik} - d_{kl}}{2}; \frac{d_{il} - d_{kl}}{2}; |d_{ij} - M|\right\} \text{ denoted } L^2_{ijkl}\cdot$$

Due to the choice of one case among cases 1 and 2, we take the minimum of L^1_{ijkl} and L^2_{ijkl}. Thus:

$$\sum_{t=1}^{n} r_t \geq \min\left\{L^1_{ijkl}; L^2_{ijkl}\right\}.$$

Moreover, for a given i, j, this inequality is verified. So that:

$$\sum_{t=1}^{n} r_t \geq \min_{i<j} \max_{k<l; k,l\neq i,j} \min\left\{L^1_{ijkl}; L^2_{ijkl}\right\}\cdot$$

We note $h(M) = \min\limits_{i<j} \max\limits_{k<l;k,l\neq i,j} \min\left\{L^1_{ijkl}; L^2_{ijkl}\right\}$.

By applying this bound to the tetrahedron example, the three functions are plotted. The result is shown in Figure 4.5. The lower bound is equal to 0.1276 for $M = 1.1276$ and as we have seen the minimum obtained by solving $\mathcal{P}_{r,x}$ is equal to 0.4226 so the solution of problem $\mathcal{P}_{r,x}$ is three time smaller than the bound, that is not bad.

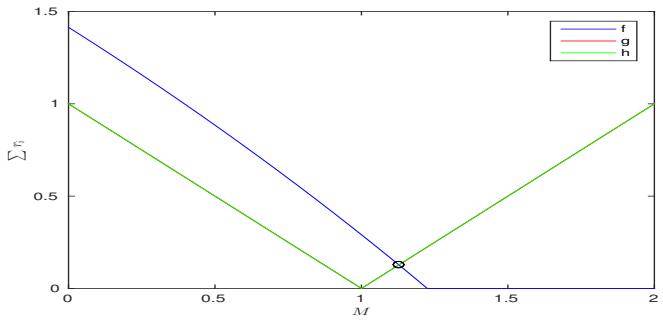


Figure 4.5: The curves of the three functions f, g and h. Functions g and h are equal due to the fact that all distances are equal to 1 . The minimal intersection is given by the black circle for $M = 1.1276$ and $\sum_{i=1}^{n} r_i > 0.1276$.

4.4 Optimization tools

4.4.1 Initialization point of problem $\mathcal{P}_{r,x}$

Different resolutions of problem $\mathcal{P}_{r,x}$ can be obtained using different initial values of matrix X. Three possible initial values can be used. The first of them is the matrix obtained by PCA or another projection method. In what follows, we present two other possibilities.

Initial point using squared distances The optimization problem $\mathcal{P}_{r,x}$ can be changed by taking the squared distances between points instead of the distances. Rewriting r_i^2 as R_i, the problem is changed into

$$\mathcal{P}_{R,x} : \begin{cases} \min\limits_{R_1,\ldots,R_n\in\mathbb{R}^+, x_1,\ldots,x_n\in\mathbb{R}^k} \sum\limits_{i=1}^{n} R_i \\ s.t. \;\; |d_{ij}^2 - \|x_i - x_j\|^2| \leq R_i + R_j, \quad \text{for } 1 \leq i < j \leq n. \end{cases}$$

The transformation is interesting as if the constraints of problem $\mathcal{P}_{R,x}$ are satisfied, the constraints of problem $\mathcal{P}_{r,x}$ will also be satisfied. Indeed:

$$\begin{aligned} & |d_{ij}^2 - \|x_i - x_j\|^2| \leq R_i + R_j = r_i^2 + r_j^2 \\ \Rightarrow \quad & (d_{ij} - \|x_i - x_j\|)\,(d_{ij} + \|x_i - x_j\|) \leq r_i^2 + r_j^2 \leq (r_i + r_j)^2 \\ \Rightarrow \quad & |d_{ij} - \|x_i - x_j\||^2 \leq (r_i + r_j)^2 \\ \Rightarrow \quad & |d_{ij} - \|x_i - x_j\|| \leq (r_i + r_j)\cdot \end{aligned}$$

That way problem $\mathcal{P}_{R,x}$ can serve as an initial step for solving problem $\mathcal{P}_{r,x}$.

Initial point using improved solution of problem $\mathcal{P}_r$ First, we give two properties which provide a way to improve the optimization results of problem $\mathcal{P}_{r,x}$.

Property 4.2. *Let us consider a point x_i such that for an index j, the following inequality is saturated :*

$$|d_{ij} - \|x_i - x_j\|| \leq r_i + r_j,$$

and the other inequalities involving i are not saturated. Then, the corresponding solution can be improved by moving x_i along the direction $x_j - x_i$ to decrease r_i and $|d_{ij} - \|x_i - x_j\||$.

Proof. The above condition means that x_i is rewritten $x_i + a(x_j - x_i)$ with $a \in \mathbb{R}$ and we look for a such that $|d_{ij} - \|x_i + a(x_j - x_i) - x_j\|| < r_i + r_j$. In particular $a \leq 0$ if $d_{ij} - \|x_i - x_j\| \geq 0$ and $a > 0$ otherwise. Let us now consider the other inequalities corresponding to index pairs (i,k) with $k \neq j$. For each of them, either $\exists a \in [a_k^{'}, a_k^{''}]$ with $a_k^{'} < 0$ and $a_k^{''} > 0$ such that

$$|d_{ij} - \|x_i + a(x_j - x_i) - x_j\|| \leq r_i + r_j,$$

as these constraints are unsaturated. Finally, if we take a different from 0 in $[a^{'}, a^{''}]$ with $a^{'} = \max_k a_k^{'}$ and $a^{''} = \min_k a_k^{''}$, all constraints involving i get unsaturated so that r_i can be decreased, decreasing so the objective function. Depending on whether a must be negative or positive, we take $a = a^{'}$ or $a = a^{''}$ respectively.

□

Another manner to improve the resolution of problem $\mathcal{P}_{r,x}$ is to effectuate a scale change by multiplying the coordinates x_i, for $i = 1, \ldots, n$, by a constant $a \in \mathbb{R}$. Thus, the new optimization problem is given by:

$$\mathcal{P}_{r,a} : \begin{cases} \min\limits_{r_1,\ldots,r_n,a \in \mathbb{R}^+} \sum\limits_{i=1}^{n} r_i \\ s.t. \;\; |d_{ij} - a\|x_i - x_j\|| \leq r_i + r_j \end{cases}$$

Property 4.3. *Let $r_1, \ldots, r_n; x_1, \ldots, x_n$ be a feasible solution of $\mathcal{P}_{r,x}$, if $\exists a$ such that $\eta(a) < \sum\limits_{i=1}^{n} r_i$ with $\eta(a) = \sum\limits_{1 \leq i < j \leq n} |d_{ij} - a\|x_i - x_j\||$, then $\exists\ \tilde{r}_1, \ldots, \tilde{r}_n$ a solution of $\mathcal{P}_{r,a}$ such that $\sum\limits_{i=1}^{n} \tilde{r}_i < \sum\limits_{i=1}^{n} r_i$.*

Proof. Let us consider $r_1, \ldots, r_n; x_1, \ldots, x_n$ a feasible solution of problem $\mathcal{P}_{r,x}$ and $a, \tilde{r}_1, \tilde{r}_2, \ldots, \tilde{r}_n$ the optimal solution of $\mathcal{P}_{r,a}$. For the solution of $\mathcal{P}_{r,a}$, for each point i, we have a certain saturated constraint associated to point k noted $C_{ik(i)}$, otherwise it would not be an optimum. So, we have:

$$\begin{gathered}
|d_{i1} - a\|x_i - x_1\|| \leq \tilde{r}_i + \tilde{r}_1 \\
\vdots \\
|d_{ik(i)} - a\|x_i - x_{k(i)}\|| = \tilde{r}_i + \tilde{r}_{k(i)} \\
\vdots \\
|d_{ij} - a\|x_i - x_j\|| \leq \tilde{r}_i + \tilde{r}_j \\
\vdots \\
|d_{in} - a\|x_i - x_n\|| \leq \tilde{r}_i + \tilde{r}_n.
\end{gathered}$$

Then, $|d_{ik(i)} - a\|x_i - x_{k(i)}\|| = \tilde{r}_i + \tilde{r}_{k(i)} \geq \tilde{r}_i$. By summing all points i, for $i = 1, \ldots, n$, we obtain:

$$\sum_{i=1}^{n} |d_{ik(i)} - a\|x_i - x_{k(i)}\|| \geq \sum_{i=1}^{n} \tilde{r}_i.$$

Thus

$$\sum_{1 \leq i < j \leq n} |d_{ij} - a\|x_i - x_j\|| \geq \sum_{i=1}^{n} |d_{ik(i)} - a\|x_i - x_{k(i)}\|| \geq \sum_{i=1}^{n} \tilde{r}_i.$$

Note $\eta(a) = \sum_{1 \leq i < j \leq n} |d_{ij} - a\|x_i - x_j\||$, then if $\eta(a) < \sum_{i=1}^{n} r_i$ there is a solution of $\mathcal{P}_{r,a}$ such that $\sum_{i=1}^{n} \tilde{r}_i < \sum_{i=1}^{n} r_i$. □

The new initial point is then given by using these two properties as follows:

- firstly, improve the solution of $\mathcal{P}_r$ using property 4.3 by solving $\mathcal{P}_{r,a}$.
- secondly, improve the solution of $\mathcal{P}_{r,a}$ using property 4.2.

4.4.2 Algorithm 1

Using the different initial values of matrix X presented above, we solve now problem $\mathcal{P}_{r,x}$. For this task, we introduce a new algorithm denoted algorithm 1 which gives the best solution that can be obtained using the different initial values cited above. This algorithm is consisted of two steps: initialization step and optimization step and it is presented as follows:

Algorithm 3

Input: D: distance matrix, N: number of iterations.

Initialization step

Project the points using PCA or MDS.

Solve $\mathcal{P}_r$ using an interior-point method. Obtained solution: $(X_{\mathcal{P}_r}, r_{\mathcal{P}_r})$.

Solve $\mathcal{P}_{R,x}$ using an active-set method and starting from the solution of $\mathcal{P}_r$ obtained at the previous step. Obtained solution: $(X_{\mathcal{P}_{R,x}}, R_{\mathcal{P}_{R,x}})$.

$X_0 \leftarrow X_{\mathcal{P}_{R,X}}$.

for $t = 1$ to N **do**

Solve $\mathcal{P}_{r,a}$ starting from X_0 using an interior-point method.

Improve the solution of $\mathcal{P}_{r,a}$ using property 1. Obtained solution: $(X^I_{\mathcal{P}_{r,a}}, r^I_{\mathcal{P}_{r,a}})$.

$X_0 \leftarrow X^I_{\mathcal{P}_{r,a}}$.

end for

Optimization step

Optimize $\mathcal{P}_{r,x}$ using an active-set method and starting from X_0, $X_{\mathcal{P}_r}$ and $X_{\mathcal{P}_{R,x}}$.

Choose the minimal solution obtained by these three different starting points.

4.4.3 Algorithm 2

Problem $\mathcal{P}_{r,x}$ is a hard problem, so it is natural to resort to stochastic optimization methods. In the present case, Metropolis-Hastings algorithm [11] allows us to build a Markov chain with a desired stationary distribution. The only delicate part is the choice of the proposal distribution and the necessity to solve a $\mathcal{P}_r$ problem at each iteration. In details, this Metropolis-Hastings algorithm requires:

1- A *target distribution*:

The target distribution is related with the objective function of problem $\mathcal{P}_{r,x}$ and it is given by:

$$\pi(s) \propto \exp\left(\frac{-E(x)}{T}\right),$$

with E an application given by:

$$\begin{array}{rccl} E: & \mathbb{R}^n & \longmapsto & \mathbb{R} \\ & x = (x_1, \ldots, x_n) & \longmapsto & E(x) = \text{Solution of problem } \mathcal{P}_r \text{ with } x \text{ fix.} \end{array}$$

The variable T is the temperature parameter, to be fixed according to the value range of E.

2- A *proposal distribution*:

The choice of the proposal distribution is very important to obtain interesting results. It should be chosen in such a way that the proposal distribution gets close to the target distribution. The proposal distribution $q(X \to .)$ has been constructed as follows, giving priority to the selection of points involved in saturated constraints:

– For each point i, choose a point $j^{(i)}$ with probability equal to:

$$P_{j^{(i)}} = \frac{\lambda \exp\left(-\lambda(r_i + r_{j^{(i)}} - |d_{ij^{(i)}} - \|x_i - x_{j^{(i)}}\|)\right)}{\sum_{k=1,k\neq i}^{n} \lambda \exp\left(-\lambda(r_i + r_k - |d_{ik} - \|x_i - x_k\|)\right)}.$$

– Choose a constant $c_{ij^{(i)}}$ using Gaussian distribution $\mathcal{N}_k(0, \sigma)$.

– Generate a matrix X^* by moving each vector x_i of matrix X^{t-1} as follows:

* If $d_{ij^{(i)}} - \|x_i - x_{j^{(i)}}\| > 0$ then $x_i^* = x_i + |c_{ij^{(i)}}|L_i$.
* else $x_i^* = x_i - |c_{ij^{(i)}}|L_i$.

with $L_i = \dfrac{x_i - x_{j^{(i)}}}{\|x_i - x_{j^{(i)}}\|}$.

3- A *linear optimization problem*:

For the matrix X generated in each iteration, we solve the linear optimization problem $\mathcal{P}_r$.

4.5 Numerical application

The presented projection method has been applied to different types of real data sets so as to illustrate its generality.

4.5.1 The data

Four real data sets are used and divided into three categories:

- Quantitative data: Iris and car data sets.
- Categorical data: Soybean data set.
- Functional data: Coffee data set.

The Iris data set [1] is a famous data set and is presented to show that the projection is as expected. This data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. The four variables studied in this data set are: sepal length, sepal width, petal length and petal width (in *cm*). Car data set [15] is a data set studied in the book of Saporta (Table 17.1, page 428). This data set describes 18 cars according to various variables (cylinders, power, length, width, weight, speed).
The soybean data set [16] from *UCI Machine Learning Repository* characterizes 47 soybean disease case histories defined over 35 attributes. Each observation is identified by one of the 4 diseases: Diaporthe Stem Canker (D1), charcoal Rot (D2), Rhizoctonia Root Rot (D3) and Phytophthora Rot (D3).

The coffee data set is a time series data set used in chemometrics to classify food types. This kind of time series is seen in many applications in food safety and quality insurance. This data set is taken from *UCR time Series Classification and Clustering* website [7]. Coffea Arabica and Coffea Canephora variant Robusta are the two species of coffee bean which have acquired a worldwide economic importance and many methods have been developed to discriminate between these two species by chemical analysis [6].

4.5.2 Experimental setup

In practice, we have tested our method on the different data sets by solving the optimization problem $\mathcal{P}_{r,x}$ using algorithm 1 and also the proposed Metropolis-Hastings algorithm (algorithm 2). Each time, a distance matrix is required. For the quantitative data, we compute the Euclidean distance between points y_i, for $i = 1, \ldots, n$, by the known formula $d_{ij} = \sqrt{\sum_{k=1}^{p}(y_{ik} - y_{jk})^2}$. For categorical data, the distance between two soybean diseases (i, j) is given through Eskin dissimilarity (or proximity) measure [5] computed by the formula $p_{ij} = \sum_{t=1}^{Q} w_t p_{ij}^t$ where $p_{ij}^t = \begin{cases} 1 & \text{if } i^t = j^t \\ \dfrac{n_k^2}{n_k^2 + 2} & \text{else} \end{cases}$, p_{ij}^t is the per-attribute Eskin dissimilarity between two values for the categorical attribute indexed by t, w_t is the weight assigned to the attribute t, Q is the number of attributes and n_t is the number of values taken by each attribute. Then, using the formula which transforms the dissimilarity into similarity: $p_{ij} = 1 - s_{ij}$, the distances can be given by the standard transformation formula from similarity to distance: $d_{ij} = \sqrt{s_{ii} - 2s_{ij} + s_{jj}}$. On top of that, to compute the distances between the curves of functional data, we have chosen a measure of proximity similar to that studied in [10]. In this article, the authors develop a proper classification designed to distinguish the grouping structure by using a functional k-means clustering procedure with three sorts of distances. So, in our work we choose one of these three proximity measures forasmuch their results are similar. Thus, the proximity measure chosen between two curves F_i and F_j is the following: $d_0(F_i, F_j) = \sqrt{\int_{\mathcal{T}} (F_i^0(t) - F_j^0(t))^2 dt}$. This measure is calculated using the function *metric.lp()* of the *fda.usc* package for the **R** software.

To solve the different optimization problems, we have used the optimization toolbox in MATLAB. For problems $\mathcal{P}_r$ and $\mathcal{P}_{r,a}$, we apply firstly PCA – for quantitative data – or MDS – for categorical and functional data – and then a linear programming package is used to solve the optimization problems using an interior-point algorithm. Problems $\mathcal{P}_{r,x}$ and $\mathcal{P}_{R,x}$ are non-linear optimization problems, therefore we use a non-linear programming package to solve it selecting the active-set algorithm to obtain the best values of $(x_1, \ldots, x_n)$ and $(r_1, \ldots, r_n)$. This iterative algorithm is composed of two phases. In the

Table 4.1: Optimization solution of problem $\mathcal{P}_{r,x}$ for different data sets.

	$\sum r_i^{\text{Algo 1}}$	$\sum r_i^{\text{MH}}$	Lower Bound
Iris	16.19	17.2	1.07
Cars	3.27	3.35	1.21
Soybean	3.98	3.93	0.29
Coffee	21.68	21.97	0.89

first phase (the feasibility phase), the objective function is ignored while a feasible point is found for the constraints. In the second phase (the optimality phase), the objective function is minimized while feasibility is maintained [17].

Our proposed Metropolis-Hastings algorithm can provide a good solution if parameters λ, σ and T are chosen adequately. For instance, λ should be such that the points belonging to unsaturated constraints are chosen with small probabilities. Therefore, we take it equal to 100. For the other parameters σ and T, we take their values respectively in a range from 0.01 and 100.

As we have mentioned in the section of visualization, the visualization of the projection of each point i in $\mathbb{R}^2$ is presented as a circle having x_i as center and r_i as radius so as the projected point belongs to this circle and this is the specificity of our method. For each data set, we show the circles obtained for each point after resolution of optimization problem $\mathcal{P}_{r,x}$. To compare the projection quality of our representation with that obtained by PCA, we use the squared cosine values as PCA projection quality. Furthermore, the lower bound defined in section 4.3 is each time computed.

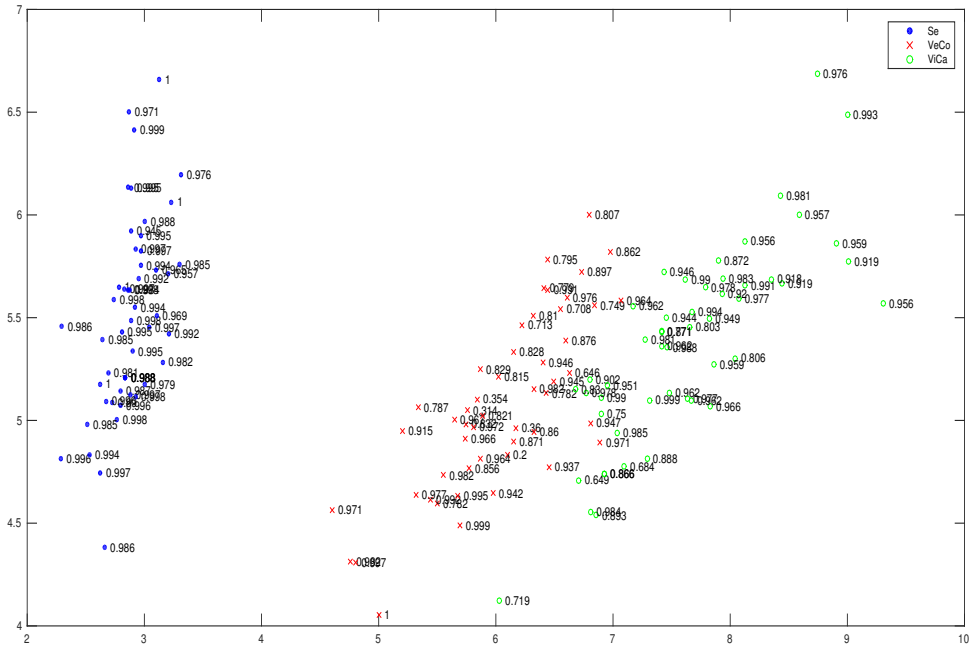
4.5.3 Results

4.5.3.1 Visualization data in $\mathbb{R}^2$

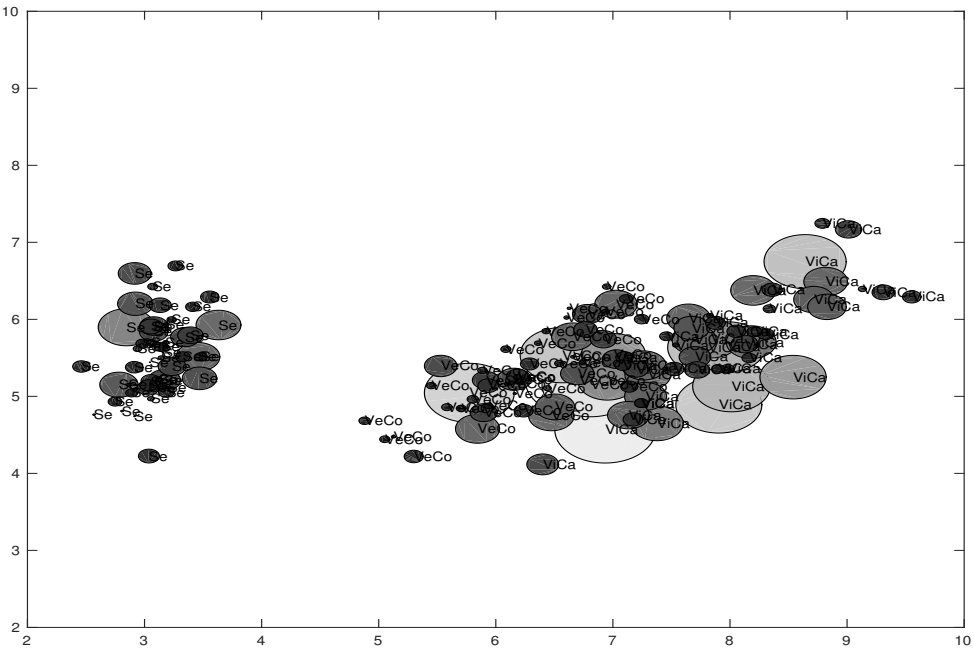
The optimization results for these four data sets are given in Table 4.1. For each data, we give the algorithm 1 and Metropolis-Hastings results with which initial starting point is used in algorithm 1. The lower bound value for each data set is also given in this table. We observe that in one case (cars), this lower bound indicates that the found solution is not far from the optimum but in the other cases, it seems that the lower bound while providing a good starting point can be improved.

Figures 4.6 and 4.7 depict the results of projection under pairwise distance control for quantitative data. This projection is compared with the projection given by PCA by plotting the projection of the points indexed by their squared cosine values.

In the projection of Iris data set showed in Figure 4.6b, it is interesting to remark that appealingly two areas are well separated. This corresponds to the well-known fact that

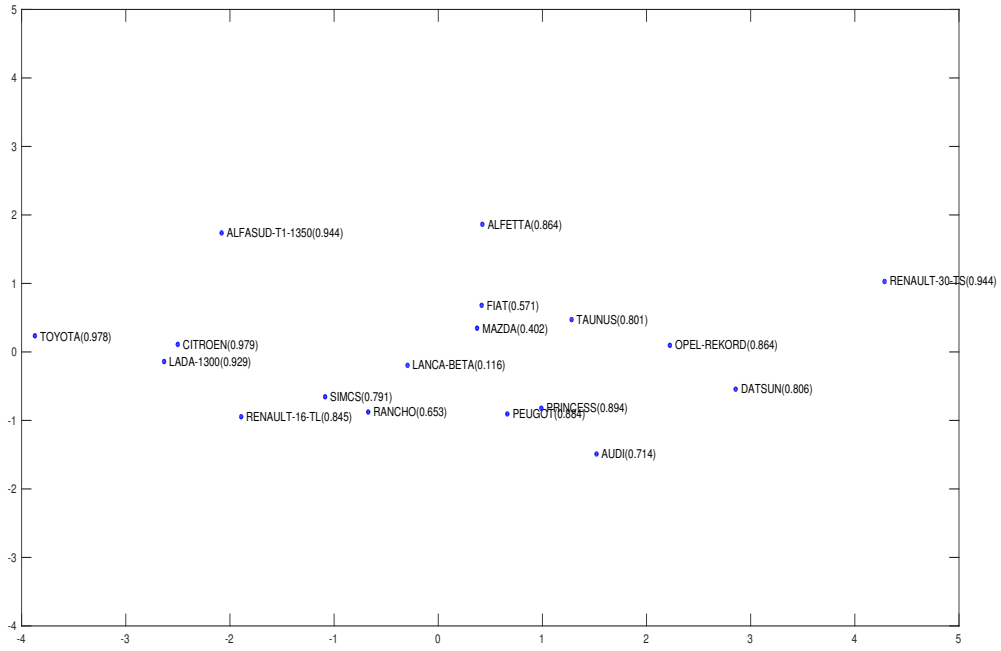


(a)

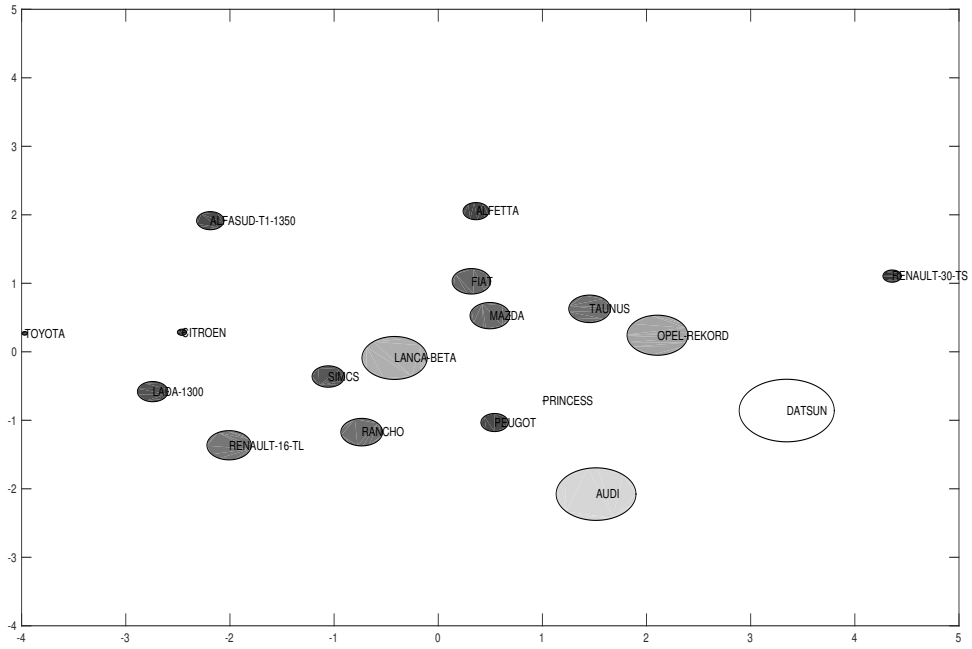


(b)

Figure 4.6: Projection of Iris data set. (a) and (b) show the projection quality using PCA and projection under pairwise distance control methods respectively. Two well separated groups can be observed.



(a)



(b)

Figure 4.7: Projection of car data set. (a) and (b) show the projection quality using PCA and projection under pairwise distance control methods respectively. For PCA, the values of the quality are given between parentheses near each car.

Iris versicolor and virginica are close whereas the species Iris setosa are more distant.

Concerning car data set, the projection of points using projection under pairwise distance control is given in Figure 4.7b. The expensive cars as "Audi 100", "Alfetta-1.66", "Dastun-200L", "Renault 30" are well-separated from the low-standard cars as "Lada-1300", "Toyota Corolla", "Citroen GS Club", "Simca 1300". We remark that the expensive cars are located on the right and low-standard one are located on the left.

By comparing the projection quality for each method presented in Figures 4.6 and 4.7 for these two data sets, we can say that our method projected the points without giving any importance to any group. Indeed, Figure 4.6a depicts a group with small values of quality measure and a group with high values of quality measure whereas the radii obtained by projection under pairwise distance control method are distributed in an equivalent way. Additionally, from Figure 4.7b, we can assert that the projected points obtained using projection under pairwise distance control method are well separated as there is any intersection between the circle. Moreover, the pairwise distances are significant in our method and give an interpretation on the position between points whereas the distances between the projected points using PCA are not interpretable as the cosine values can not be interpreted as distances. This is the particular strength of our method. Hence, projection under pairwise distance control suggests an absolute interpretation whereas PCA gives a relative one. From this, we can conclude from Figure 4.7b that there is a big difference between the two cars "Toyota" and "Renault 3" as the distances between this two cars is very important. Conversely, the distance between "Lada1300" and "Citroen" cars are small indicating then the closeness of these two cars. Note here that these two cars are very well projected leading to a very good interpretation.

For the qualitative and functional data sets, it is necessary to verify that the matrix B obtained by MDS method is semi-definite positive to use the squared cosine as quality measure because the starting point of optimization is obtained from MDS. After that, in case of positiveness of matrix B, we can calculate the quality measure. In the projection of the soybean data set, four classes have been shown in Figure 4.8 and each class contains the diseases number of the class. But basically, the whole set of points can be divided in two large classes. Indeed, It is clear that class 2 is well separated from the others classes as there is no intersection between the circles of class 2 and the circles of others classes. Moreover, class 1 can be considered as well separated class from classes 3 and 4 if we do not take into account the point D_3^*. Classes 3 and 4 are not at all well separated as we can exhibit that there are different intersections between the circles of these two classes. This result is figured in [16] which lists the value "normal" for the first two classes and "irrelevant" for the later two classes. The comparison of projection under pairwise distance control result with PCA is not possible for this data set because the matrix B is not semi-definite positive.

The coffee data set has been studied in several articles ([6, 3]) and different classification

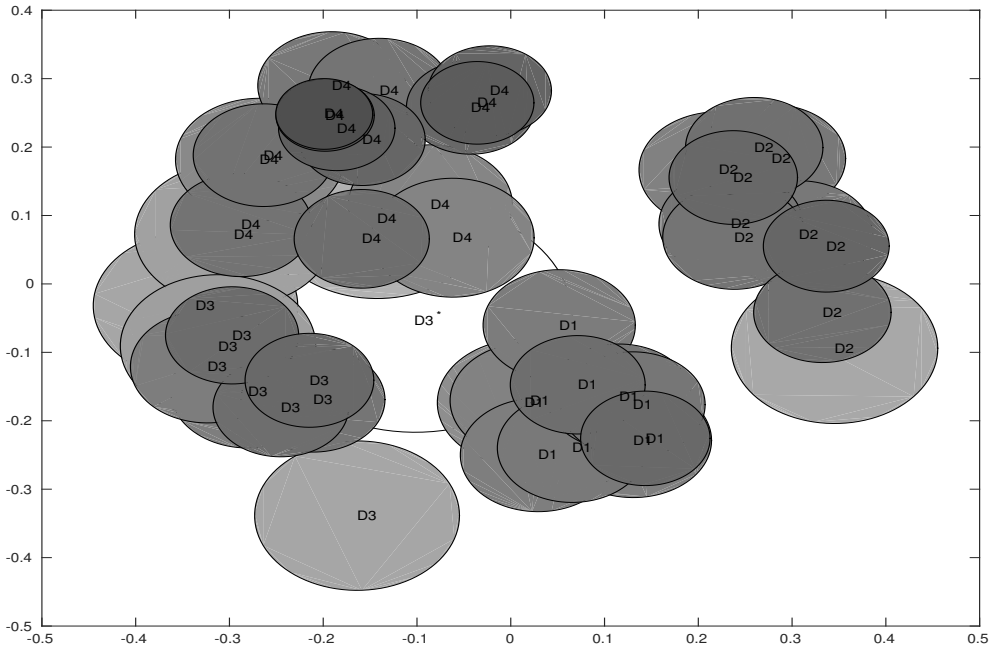
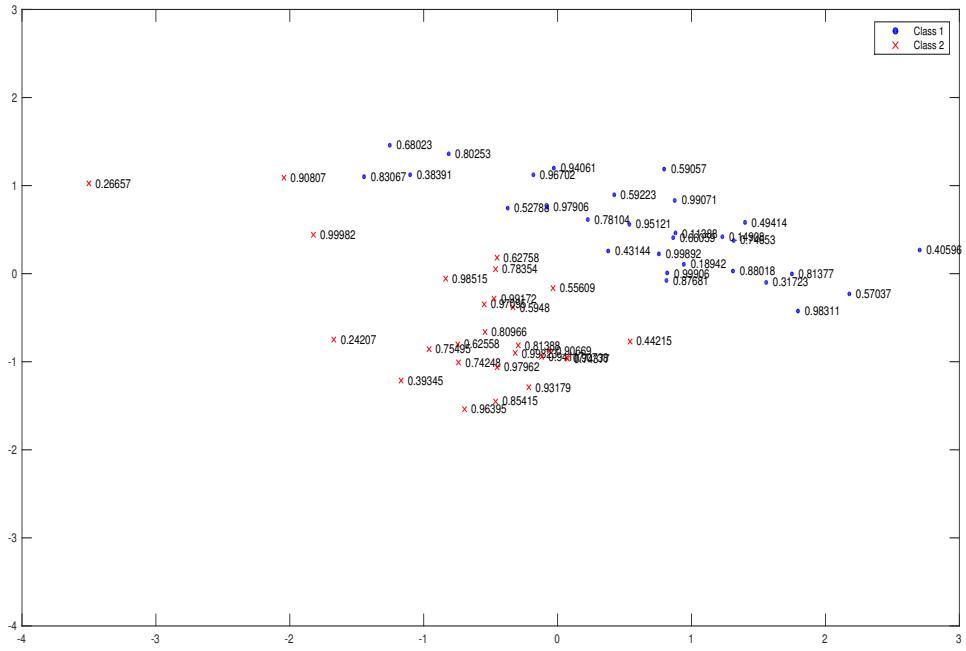


Figure 4.8: Projection under pairwise distance control for soybean dat set. Four groups are presented, indexed by D1, D2, D3 and D4.

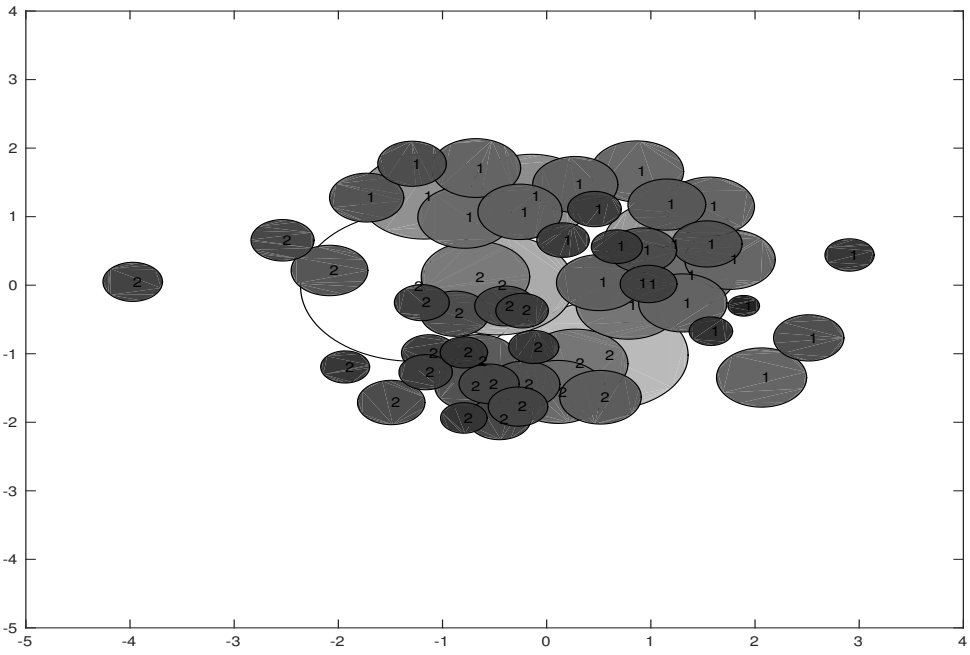
methods have shown the different groups contained in this data set using our method and PCA. We can see clearly in Figure 4.9 the grouping structure that is obtained. In Figure 4.9b, we show that we have succeeded in differentiating the Arabica from Robusta coffee. These two classes are clearly presented, the first class indexed by 1 corresponding to Arabica coffee and the second one indexed by 2 corresponding to Robusta coffee. These classes are not well separated by comparing with the results of quantitative data, since there are many intersections. Therefore, the representation of the points as circles and not as coordinates points gives more information about the real class of points and shows the points who have the possibility to be misplaced in a class.

Figure 4.9a shows the projection quality using PCA. As all the eigenvalues of matrix B are positive, so we can compute the quality measure given by PCA. Comparing the projection quality of PCA and projection under pairwise distance control given respectively by Figures 4.9a and 4.9b, we can observe that the quality of projection of the set of points is pretty steady.

Additionally, Metropolis-Hastings has been applied to these data sets. The trace plots of the optimization problem $\mathcal{P}_{r,x}$ are shown in Figure 4.10 after 5000 iterations. Returning to Table 4.1, we can exhibit that Metropolis-Hastings algorithm solutions are very close to those obtained using the optimization package of Matlab and reciprocally. Thus, the obtained radii should be close to the optimum.



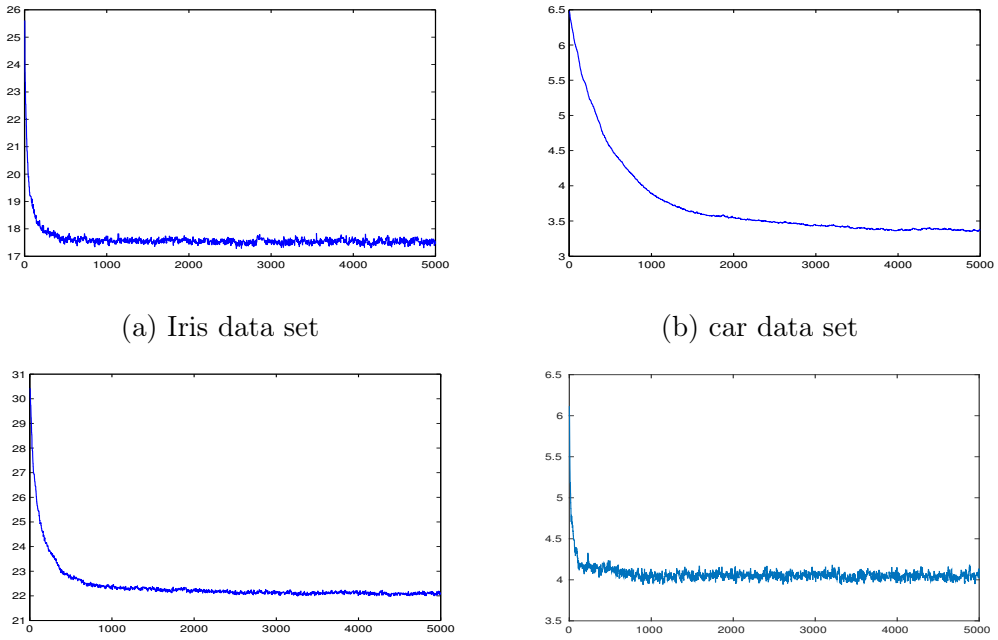
(a)



(b)

Figure 4.9: Projection of coffee data set. (a) et (b) show the projection quality using PCA and projection under pairwise distance control respectively. Two clusters indexed 1 and 2 indicate respectively Arabica and Robusta classes.

Finally, we present the lower bound computed from the three functions described in Section 3. The lower bound is taken by the minimal intersection of these functions. Returning to Table 4.1, it is clear that the value of the bound is small compared to the value of the solution obtained by algorithm 1 and Metropolis-Hastings. Thus, this bound while providing a good starting point, should be improved. Note that this bound for tetrahedron example gives also good results as algorithm 1 provides a solution three times smaller than the bound.



(a) Iris data set

(b) car data set

(c) Coffee data set

(d) Soybean data set

Figure 4.10: Trace plots of Metropolis Hastings for different data sets. The x-axis corresponds to the iteration number and y-axis to the value of $\sum_{i=1}^{n} r_i$.

4.5.3.2 Dimensionality reduction results

One of high-dimensional data studies objectives is to choose from a large number of variables those which are important for understanding the underlying phenomena of study. So, the aim will be to reduce the dimension rather than to visualize data in $\mathbb{R}^2$. So, our method can also serve to reduce the number of variables by taking into account the minimal value of $\sum_{i=1}^{n} r_i$.

Here, we have solved the problem $\mathcal{P}_{r,x}$ using the different possible dimension values. We have plotted in Figure 4.11 the values of $\sum_{i=1}^{n} r_i$ as a guide for choosing the reduced number of variables. This figure shows the values of $\sum_{i=1}^{n} r_i$ for the different data sets using different dimensions. It is clear to see that the value of $\sum_{i=1}^{n} r_i$ decreases when the dimension increases.

The main problem which is widely posed in dimension reduction methods is the determination of the number of components that are needed to be retained. Many methods have been discussed in the literature [12, 4] to determine the dimension of reduced space relying on different strategies related to the good explanation or the good prediction. So, with our method the choice of the reduced space dimension is related to the locally projection quality of points and how much the user is interested by the projection quality of points.

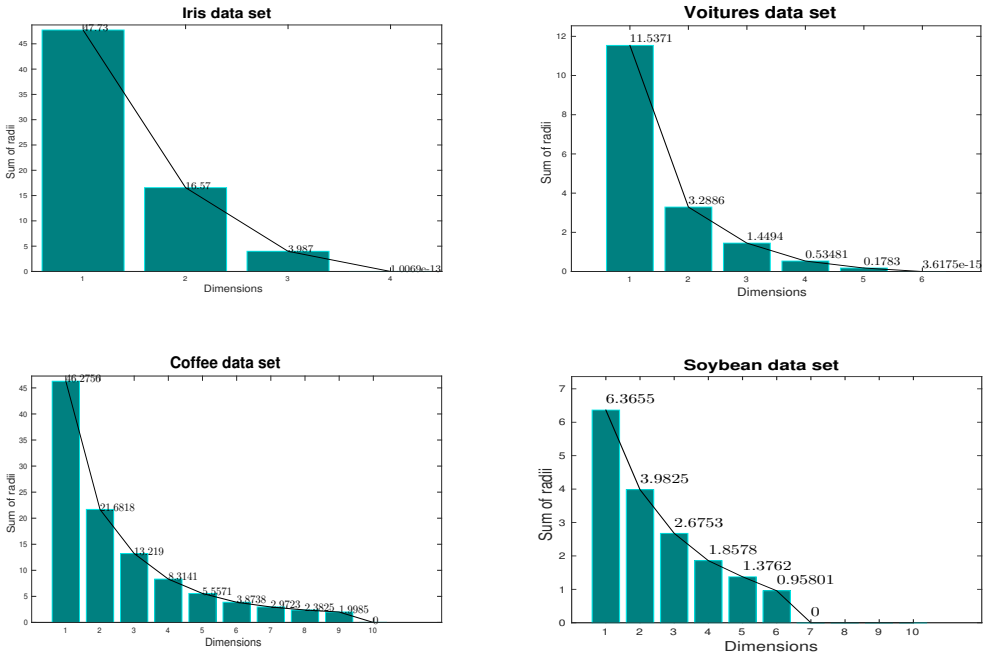


Figure 4.11: The scree plot of $\sum_{i=1}^{n} r_i$ for different dimensions for the four data sets.

Concerning the quantitative data sets (Iris and car), if the main objective of the user is to obtain a very good projection quality then a choice of three components against 4 for iris and 6 for cars can be a good choice as the value of $\sum_{i=1}^{n} r_i$ is small and there is not a big difference between this value for this dimension and the values for the higher dimensions. For coffee data set, a dimensionality reduction from 56 sample time series down to 6 simple extracted features is considered as a good choice. The same idea can be seen for soybean data set, a reduced space dimension equal to 4 can be considered as efficient reduced space.

Moreover, a comparison of our results with the existent results shows a coherence between them. For Iris data set, [8] and [13] conclude that the number of variables can be reduced to 2 as the petal length and petal width variables are the most important variables among all variables. Similarly, this result can be seen for car data set. Saporta in his book [15] (Table 7.4.1 page 178) notices that the conservation of two dimensions leads to the explanation of 88% of inertia. So, these results seem very similar to our results, the important decrease is located between dimensions 1 and 2. The other decreases are

negligible for these two data sets. Selection variables is studied on time series coffee data set in [2]. Using several analysis methods, the number of selected variables ranges between 2 and 13. This result is also seen using our method, a number of reduced variables taken between 2 and 9 gives a good quality projection of the points. Concerning soybean data set, Dela Cruz shows in his paper [9] that the 35 attributes can be reduced to 15 and here with our method, we have succeeded to reduce the attributes to 6 by having a very good projection.
Hence, the presented results confirm that we can reduce the dimension non-linearly and still keep a way of assessing as reasonable number of dimensions and that is efficient as a dimensionality reduction method.

4.5.4 Advantages of projection under pairwise distance control method

As we have seen, our presented method has several advantages. To summarize:
firstly, it is a non-linear projection method which takes into account the projection quality of each point individually. Secondly, the distances between projected points are related to the initial distances between points offering a way to interpret easily the distances observed in the projection plane. Thirdly, the quality distribution between the points seems to be evenly distributed.

4.6 Conclusion

The purpose of this chapter was to outline a new non-linear projection method based on a new local measure of projection quality. Of course, in some projection methods a local measure is given but this measure cannot be applied unless in cases of linear projections, and even then it is not suitable for graphical representation.
The quality of projection is given here by additional variables called radii, which enable to give a bound on the original distances. We have shown that the idea can be written as an optimization problem in order to minimize the sum of the radii under some constraints. As the solution of this problem cannot be obtained exactly, we have developed different algorithms and proposed a lower bound for the objective function.

Bibliography

[1] Anderson E, The Irises of the Gaspé Peninsula, Bull. Am. Iris Soc., 59(1935), 2–5.

[2] Andrews, J. L. and McNicholas, P.D., Variable Selection for Clustering and Classification, Journal of Classification, 31(2014), 136–153.

[3] Bagnall, A., Davis, L., Hills, J., and Lines, J., Transformation Based Ensembles for Time Series Classification, Proceedings of the 12th SIAM International Conference on Data Mining(2012), 307–319.

[4] Besse, P., PCA stability and choice of dimensionality, Statistics & Probability Letters, 13(1992), 405–410.

[5] Boriah, S., Chandola, V., and Kumar, V., Similarity Measures for Categorical Data: A Comparative Evaluation, Proceedings of the SIAM International Conference on Data Mining(2008).

[6] Briandet, R., Kemsley, E. K., and Wilson, R. H., Discrimination of arabica and robusta in instant coffee by fourier transform infrared spectroscopy and chemometrics, J. Agric. Food Chem., 44 (1)(1996), 170–174.

[7] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A. and Batista, G., The UCR Time Series Classification Archive, `www.cs.ucr.edu/~eamonn/time_series_data/`(2015).

[8] Chiu, S. L., Method and Software for Extracting Fuzzy Classification Rules by Subtractive Clustering, In Proceedings of North American Fuzzy Information Processing Society Conference(1996).

[9] Dela Cruz, G. B., Comparative Study of Data Mining Classification Techniques over Soybean Disease by Implementing PCA-GA, International Journal of Engineering Research and General Science, 3(5)(2015), 6–11.

[10] Ieva, F., Paganoni, A.M., Pigoli, D., and Vitelli., V., Multivariate functional clustering for the analysis of ECG curves morphology, Journal of the Royal Statistical Society, Applied Statistics, series C., 62(3)(2012), 401–418.

[11] Johansen, A. M. and Evers, L. Monte Carlo Methods, Department of Mathematics, University of Bristol (2007).

[12] Jollife, I.T., Principal Component Analysis, Springer, New York (1986)

[13] Liu, H., Setiono, R., Chi2: feature selection and discretization of numeric attributes, In Proceedings., Seventh International Conference on Tools with Artificial Intelligence (TAT'95) (1995).

[14] Mardia, K.V., Kent, J.T. and Bibby, J.M., Multivariate analysis, Academic Press, London (1979).

[15] Saporta, G., Probabilités, analyse des données et statistique, Technip (2006).

[16] Stepp, R., Conjunctive conceptual clustering, Doctoral dissertation, department of computer science, university of Illinois, Urbana-Champaign, IL (1984).

[17] Wong, E., Active-Set Methods for Quadratic Programming, Ph.D. thesis, university of California, San Diego (2011).

Conclusion

This thesis contributes to provide two new multivariate data analysis methods. Specifically, this thesis deals with the dimensionality reduction and visualization.

Fitting distances was the motivation of the MDF method presented in chapters 2 and 3. We considered two matrices, target and reference matrices, which the proposal method fit the distance matrix computed after modification of the coordinates of the target matrix to the reference matrix containing pairwise distance and resulting then an optimization problem where the objective function is the mean square error. Moreover, to avoid unnecessary modifications, we have added a combined penalty term to the mean square error and we have chosen a good regularization parameter to obtain good results. At this stage, a real application coming from molecular biology has been treated. Two different conformations, before and after some biological reaction, for a same protein have compared in order to detect the amino acids that undergo an important movements after the reaction. Penalized MDF method allowed to identify the parts of the protein which have moved significantly between the two conformations.

Furthermore, in chapter 3 we have introduced the random effect to the modification vectors and the objective function is not the mean square error as this quantity is a random value. The modification vectors are obtained in one hand by minimizing the expectation of the mean square error and on the other hand by simulating of the error. A statistical test has been introduced to assess how much the transformation is significative.
The objective function of the minimization problem for the random model of MDF is obtained using the expectation of the non-central chi and chi-squared distributions. Concerning the simulation, we have used Metropolis-Hastings and for that, we have developed a good proposal distribution adapted to our problem. The application of this method in the sensometrics domain shows the simplest explanation of the sensory profiles of products according the consumers preference.

Chapter 4 is dedicated to present a new projection paradigm focused on non-linear projection and takes into account the local quality projection. Projection under pairwise distance control is the method presented in this chapter, it offers a straightforward

visualization and interpretation of data in the reduced space as the distances between the projected points is related to the initial distances. The problem of this method is written as an optimization problem and different ways have been developed to solve this problem by using two different algorithms and a lower bound of the objective function. Moreover, we show that this method can be applied to reduce the dimension non-linearly and still keep a way of assessing a reasonable number of dimensions which is efficient as a dimensionality reduction method. The application of this method have made on quantitative, qualitative and functional data and the results are compared to those obtained using PCA to show that local projection quality is distributed evenly between the points and the interpretation is very easy compared with PCA as this quality can be interpreted as distances.

Perspectives

The work presented in this thesis has an interesting potential for future research:

- Perspectives on chapters two and three:

  In chapters 2 and 3,we have presented the Multidimensional Fitting method that transforms a target matrix to make it fit to a reference matrix by using deterministic and random models.

  MDF method in their penalized deterministic model and random model has been applied to quantitative data set. More studies can be developed to adapt MDF method to qualitative and functional data.

  Moreover, an interesting problem posed by biologists is a topic of further research that can be inspired from MDF method. Indeed, three kinds of distance matrix are given in order to describe the distances between the populations: the resistance distance, the geographical distance and the graph distance. The idea is to find the best distance matrix that fits better the genetic distances between populations. As first idea, we want to compare the Laplacian matrix for each distance matrix graph by fitting the Laplacian matrix of the genetic distance matrix, using the phylogenetic tree, to the Laplacian matrix of the other distances matrices. This project is proposed by *Jean-François Arnaud* from "laboratoire de génétique et évolution des populations végétales" and *Vincent Castric* from "laboratoire évolution, écologie et paléontologie" at Lille university.

- Perspectives on chapter four:

  In chapter 4, we have presented a new non-linear projection method that takes into account the local projection quality. Projection under pairwise distance control method used a lower bound to assess how good a solution is. We have seen that the lower bound values obtained on the application is not very good so further research is needed to improve the lower bound in order to assess how close the algorithms are from the minimum.

  Besides, data visualization is an important step in many studies specially with big data. Using our method with these data makes the resolution of the optimization

problem computationally costly. As a perspective, further studies will be needed to simplify the optimization and reduce the computing time. In this topic, "*Trans-manche knowledge*" is a big data project to predict the risk of flow problems in the port of Calais in function of time. The data used in this project is hour by hour data for each day over several years. The visualization of these data is important to understand more the underlying phenomenons. This work could be the starting point to understand the data.