Statistical models on manifolds for anomaly detection in medical images

Florian Tilquin

Université de Strasbourg

UNIVERSITÉ DE STRASBOURG

msii ÉCOLE DOCTORALE

ÉCOLE DOCTORALE MSII

Laboratoire ICube - UMR 7357

THÈSE

présentée par :

Florian TILQUIN

soutenue le :

13 Novembre 2019

pour obtenir le grade de : **Docteur de l'université de Strasbourg**
Discipline/Spécialité : Signal, image, automatique, robotique

Statistical Models on Manifolds for Anomaly Detection in Medical Images

THÈSE dirigée par :

PR. HEITZ Fabrice	Directeur de thèse, université de Strasbourg
PR. NAMER Izzie	Directeur de thèse, université de Strasbourg

RAPPORTEURS :

COLLIOT Olivier	Directeur de recherche CNRS, Laboratoire ARAMIS
THIRION Bertrand	Directeur de recherche INRIA, CEA Paris-Saclay

---

AUTRES MEMBRES DU JURY :

BERGER Marie-Odile	Directrice de recherche INRIA, Laboratoire LORIA
FAISAN Sylvain	Maître de conférences HDR, Laboratoire ICube

Contents

Acknowledgements

Avant toutes choses, je souhaite remercier mes rapporteurs Olivier Colliot et Bertrand Thirion, qui ont accepté de scruter ma thèse dans les moindres détails pour en tirer le meilleur, et Marie-Odile Berger qui a accepté d'être examinatrice de ce travail ainsi que présidente de ce jury.

Je remercie bien sûr Fabrice Heitz de m'avoir accueilli et encadré dès le stage de master jusqu'au bout de cette thèse sur pratiquement quatre années de recherche pendant lesquelles j'ai beaucoup appris et, je l'espère, mûri. Du fond du coeur, je souhaite aussi remercier Sylvain Faisan, que j'ai sorti de la quiétude de sa soutenance d'HDR pour l'entraîner dans le bazar qu'a pu représenter cette thèse par moment. Il a entendu mon appel à l'aide et y a répondu avec un investissement plus grand encore que s'il avait été impliqué dans cette thèse dès son commencement. Je lui dois ce document et ce diplôme.

Nombreux ont été ceux à m'encourager et à m'enjoindre de persévérer au cours de ces 3 ans, à croire en mes résultats plus que je n'y ai cru moi-même et à trouver cette thèse plus spéciale que je ne l'ai trouvée, et sans eux ce travail aurait été de bien plus mauvaise qualité.

Je commencerai donc par remercier un anonyme camarade de thèse qui m'a beaucoup appris sur la gestion du stress et les différents lieux où décompresser dans Strasbourg avec lui. Il se reconnaîtra dans cette citation de M.Perrichon: "*Ah ! jeune homme !... vous ne savez pas le plaisir qu'on éprouve à sauver son semblable.*"

Le club de rhumologie improvisé dans l'équipe a bien sûr été un soutien moral essentiel pendant ces rugueux hivers, et je remercie tout particulièrement Céline et Vincent pour leur aide et leur classe réputée jusqu'aux Amériques. Je n'oublie pas non plus Jean, l'autre Jean (mon préféré), Adrien et Denis, autres fameux partenaires de dégustations ou de jeux de société ! Cette thèse a été l'occasion de rencontrer une équipe formidable, dont je souhaite remercier tous les membres, qui m'ont chaleureusement accueillis.

À tous les doctorants encore en thèse dans l'équipe: Argheesh, Hugo, Étienne, Éléonore, Jennifer et Cyril, je vous souhaite une fin de thèse paisible et fructueuse ! Force et honneur. Nastya bien sûr, indispensable compagnon d'infortunes durant ces trois années mais surtout d'innombrables moments de rire et de détente, je serai là à ta soutenance pour te rappeler que je ne travaille qu'avec les meilleurs, et te dire que tu as eu raison de persévérer !

Il me faut bien sûr remercier mes camarades de jeux qui ont rythmés les semaines et ont permis de faire de vraies pauses vis-à-vis de la thèse. La team Korvosa et la team Dybilal (un certain Quentin étant doublon et à l'occasion un très bon ami).

Ces trois années on été riches en événements personnels, avec de nombreux changements: devenu alsacien par le sol, et breton par alliance, il m'arrive parfois de me perdre un peu dans mes orgines. Mais heureusement il me reste mes amis parisiens (Bob, Jack, Johny et Lewis) pour me faire garder

les pieds sur terre: ma famille, ma vocation, mon port d'attache. Ils essaient sans relâche de soigner mon syndrome de l'imposteur et de me faire devenir millionaire. Spécialement Jack, qui m'a appris l'ambition et sans qui je n'aurai probablement fait les études que j'ai faites. Je remercie d'ailleurs Sandrine et Étienne pour m'avoir fait traverser l'ENS de façon si plaisante et d'être encore des amis si proches. De véritables modèles !

Enfin il me reste à remercier ma famille: mes parents et mon frère qui m'ont vu comme un chercheur confirmé dès le moment où j'ai commencé cette thèse, et bien sûr ma femme Hélène, j'espère être la moitié de la personne que je vois dans ses yeux !

À toutes les personnes que j'ai pu croiser durant ces années et qui m'ont aidées par une parole, un conseil, ou par leur amitié, merci.

Je me ferai savant en la philosophie
En la mathématique et médecine aussi:
Je me ferai légiste, et d'un plus haut souci
Apprendrai les secrets de la théologie.

– Joachim du Bellay

Nomenclature

X	Training set/Learning dataset[1] with high dimension
X_n	n-th element (sample) of X
x	Low-dimensional embedding for X
$\mathcal{M}$	High-dimensional manifold subspace
$\mathcal{E}$	Low-dimensional embedding obtained from $\mathcal{M}$
N_s	Number of samples for the training set X
d	Dimension of the ambient space for X
m	Dimension of the subspace/manifold/embedding
Y	A test sample (or a set of test samples)
y	Low-dimension equivalent of Y
R_p	Residual of order p for a projection algorithm
$R_{p,n}$	Residual of order p for the n-th sample of the dataset
A^T	Transpose of matrix A
A^{-1}	Inverse of matrix A
$\overline{A}$	Sample-wise mean of A
σ_A^2	Unbiased sample-wise variance of set A
$\tilde{A}$	$A - \overline{A}$
$< u, v >$	Dot product of two vectors u and v
$u \perp v$	u and v are othogonal
Projection	Getting a sample **from high dimension to high dimension**
μ	Projection in HD ($\mu : \mathbf{R}^d \mapsto \mathbf{R}^d$)
π	Dimension reduction function ($\pi : X \mapsto \mathbf{R}^m$)
ρ	Reconstruction function ($\rho : \mathbf{R}^m \mapsto \mathbf{R}^d$)
$\tilde{\pi}$	Out-of-sample extension of $\pi : \mathbf{R}^d \mapsto \mathbf{R}^m$
$\|u\|_p$	L_p norm of u

[1] All matrices and vectors are **column vectors**, with as many columns as the number of samples

K	Kernel or Gram matrix of a set ($K \in R^{N_s \times N_s}$)
$(X_n)_s$	s-th pixel of image X_n
r	Number of regressors
V	Validation set
$\#S$	Cardinal of a set S
$\mathcal{N}_X(Y)$	Neighborhoud of Y in the set X
$A \succcurlyeq 0$	Element-wise positivy of A: $\forall s, A_s \leq 0$
D	A design matrix
β	A vector of regressors
ε	A noise variable
$\mathbb{1}_N$	Column vector of ones of size N
I_N	Identity matrix of size N
$\mathcal{S}^m$	The m-dimensional sphere

Chapter 0

Introduction

0.1 General Introduction

Throughout this thesis, we considered the problem of anomaly detection in neuroimaging data, in the context of comparing a single subject to a normal control group. Anomaly detection in medical images is a task focused on detecting abnormal patterns in images, rather than whether the image itself is abnormal or not. Most standard algorithms of anomaly detection are usually performing a one-class classification problem, detecting outliers with respect to the distribution of normal subjects. Our interest is not to make a global statement about the sample, but instead to provide a spatial localization of abnormal patterns within the subject's image data by creating a model for healthy samples, that can be applied to test images in order to evaluate whether they conform to the model or not. This task is usually done on sight by medical experts, but given the 3D nature of these data, and the small size of areas to be detected, this can be both an arduous and cumbersome task. This work is thus aimed at providing helpful detections for the experts, to guide them in their diagnostic. Figure 1 illustrates such type of detections.

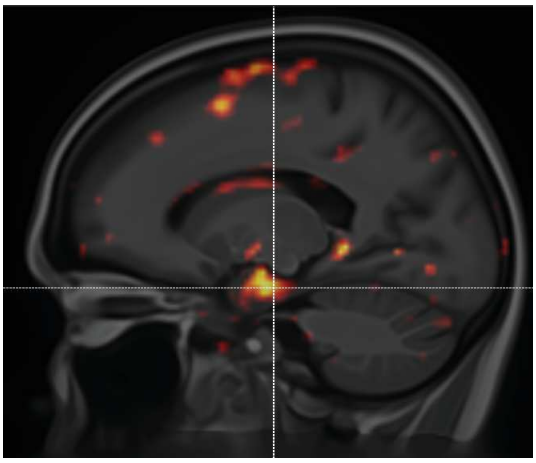

Figure 1: Illustration of an anomaly detection over a single subject over a sagittal slice. Red and yellow areas are detected as abnormal.

This thesis is based on the works of several previous PhD students, the one we put our main focus on being Torbjørn Vik [1]. In his thesis, he was already focused on performing anomaly detection using linear multivariate models, although on different medical datasets. Our goal was to go one step further in this modelization, while keeping an application in medical images. At Vik's time, the most widely used tool for performing anomaly detection in medical images was the software *Statistical Parametric Mapping*, and it still is nowadays. SPM is an univariate, linear model, making strong assumptions on the distribution of the analysed dataset. Its popularity is mostly due to its ability to provide a fast, reliable detection, based on strong statistical foundations even with large datasets. The main use of SPM by doctors is to perform group

versus group analysis, allowing them to better apprehend the two populations differences or equivalently the effect of the studied disease. The work presented in this document concentrates on the analysis of one subject compared to a group. While SPM does provide a framework for this type of analysis, it does have the same drawbacks than his group vs group counterpart. We thus defined some quite simple specifications for the methods we wanted to develop. These methods should be:

- Non-linear, multivariate methods, to take into account spatial correlations in our data;
- Do not make assumption on the data distribution.

As we introduce these methods, we should clarify that even though we worked with a specific application to an Alzheimer's Disease dataset, our goal here is not to perform prediction on some test subjects images (as to whether or not they are -or will be- afflicted with the disease), as this diagnostic is rarely ever done by doctors solely by looking at medical images, however incriminating they appear. This anomaly detection is used in a single subject vs group context to provide an helpful preliminary indication for the doctor, to focus on some areas rather than others or that he could have missed. Obviously the developed methods could also be used in a group vs group setting.

0.2 Document Layout

This manuscript is organized as follows:

In the first chapter, after introducing the paradigm of anomaly detection that we will be using throughout all of this work, we will take a look at methods that are the state of the art methods for our whole paradigm. We will then present methods of the literature that can be used as basic building elements for the paradigm.

Moving on, the second chapter will be dedicated to introducing our own contributions to the paradigm of anomaly detection. First we will explain how all of our anomaly detection methods are actually manifold projection methods, and how the anomaly detection is rather made by statistical testing. Then we will present two "classes" of projection methods: a first based on the framework and algorithms from the first chapter, and a second one that skips one step of the previous framework and unites various methods under a kernel approach. Finally we will explain in this chapter how we managed to apply to non-linear methods the robust algorithms that are usually associated to linear methods.

The third chapter is dedicated to getting a better understanding of the problems we are faced with, how the specific properties of our datasets will impact the methods we developed, and what to expect once we deal with the

real dataset. To do so, we will present the elaboration of several synthetic datasets with controllable properties and with abnormal subjects for which a clear ground truth is known. We will then proceed to analyse the results obtained over these datasets.

A fourth chapter is focused on the study of the real dataset, and the associated results of the best methods we introduced. As we are deprived of a ground truth for the real dataset, we will therefore only present averaged results of single subjects, with an analysis of these results by doctors.

The fifth and last chapter will be the conclusion of this document.

0.3 Contributions

The contributions presented in this document can be synthesized by the following list:

- The creation of fully functional methods of manifold projection based on classical non-linear dimensions algorithm, based on previously known methods of out-of-sample extension and reconstruction.
- The introduction of fully original methods of projection *via* a new framework able to link together all *kernel* methods of dimension reduction.
- The extension of robust algorithms to all of these methods.
- The elaboration of several databases, from purely synthetics to a real medical one, and the design of several tests for the comparison of all the new methods together, along with the state of the art ones.
- Convincing, consistent results on real data, that lend themselves to medical interpretation of the different forms of the AD pathology and its evolution.

Chapter 1

Methodology

This Chapter contains:

1.1 A projection paradigm

Contents for this section

1.1.1 Theoretical paradigm

In this thesis, we aim to introduce several statistical models designed to tackle the multiple drawbacks of the global linear model. They all share common characteristics: these models are multivariate (as to take into account the obvious spatial correlations that exist in our data), non-linear (to efficiently model the dataset geometry, as we will see further on) and based on the same paradigm.

This paradigm can be synthesized as:

Given a dataset X representative of the normality against which we wish to confront a test subject Y, finding the closest image to Y that could belong to the geometric structure underlying X.

From a computer science point of view, it consists in learning a model over a globally consistent dataset of individuals. We will perform the task of learning this model by means of statistical machine learning and thereby, as for any machine learning problem, we will need to gather the most possible data with the most representativeness to obtain a model with a good understanding of the normality. Once this model is known, we need to apply it to new test samples in order to "normalize" them, *i.e.* make them look like part of the training dataset.

Before going into details, we need to introduce two key concepts who intervene while dealing with the type of application this thesis focuses on. The first one is the curse of dimensionality: it is a well-documented [2, 3] effect first described by Bellman in 1961 [4], that tends to rapidly happen as the number of dimensions of the space data lies in rises. As dimensionality grows, the sparsity of data density in the intrinsic space increases exponentially with its volume (*e.g.* to uniformly sample the n-dimensional cube with k points along each side, one would need k^n points in total to do so). This tends to transform the function approximation problem (density estimation, classification, regression, etc.) into an arduous one. Indeed, for independent identically distributed random variables, it can be shown [5] that distances between points sampled from these variables become less and less meaningful as the number of dimension increases, thus making the nearest-neighbours problem used in many of the machine learning algorithms less and less efficiently solved.

However, all hope is not lost as we are obviously not dealing with independent random variables. Furthermore, we will put to good use the second key concept inducing this paradigm: the manifold hypothesis [6]. This hypothesis (that we could easily recharacterize as an axiom in our case), states that not only are high-dimensional data such as (medical) images not "independent identically distributed random variables", but they are actually lying into a subset of the ambient space called a *manifold* (see appendix 5.1.3): a non-linear subset of the euclidean space, heavily structured and possessing its own inner dimension, that is supposed to be extremely small compared to the one of the ambient space. One crucial property of manifolds is that, although they are inherently globally non-linear spaces, they are *locally* linear, *i.e.* similar (in a diffeomorphic sense) in all points to an euclidean space whose dimension is the one of the manifold.

1.1.2 Methodology

To alleviate both previously mentioned problems, we considered dimension reduction algorithms, and especially *non-linear* ones. This kind of algorithms is made to deal with the high dimensionality of these datasets while efficiently modeling their underlying manifold.

As our paradigm for anomaly detection is to find the closest image of our test image that has a coherent geometric and topological information with the control group, we thus aim at finding what we call a *projection* function μ, that transforms any sample into a "control sample" by performing a mathematical projection over the inherent geometric structure of our control set. As previously mentioned, to learn the geometric structure of our control group while coping with the huge dimensionality of our data, we will consider the set of training samples a *training set* for dimension reduction algorithms. Hence, we will have to learn two mappings in order to obtain μ: one *dimension reduction mapping* π (or embedding function), that transforms high dimensional data points into their lower dimensional embedding, and a i*reconstruction function* ρ, that is able to provide a high dimensional sample from a low dimensional one. As a result, we get $\mu = \rho \circ \pi$.

We denote the mapping obtained by our algorithm from the original space to the dimension reduction space, or *embedding*, by π:

$$\begin{aligned} \pi \colon X \subset \mathbb{R}^d &\to \mathbb{R}^m \\ X_i &\mapsto \pi(X_i) \end{aligned} \tag{1.1}$$

where X, our sample set (control group), lies in $\mathbb{R}^d$, with $d \gg m$ (see figure 1.1 for a graphical representation).

As most dimension reduction are tools designed to be applied once to a whole dataset, they are not equipped to deal with new sample points not belonging to the training dataset X and that ought to be tested (projected) for anomalies. Thus, the embedding function π is only defined on the training

dataset X, and not on the whole space $\mathbb{R}^d$. Therefore, to find a low-dimensional representation of a test sample $Y \notin X$, we will need to find what is called an out-of-sample extension of π, denoted $\tilde{\pi}$:

$$\begin{aligned} \tilde{\pi} \colon \mathbb{R}^d &\to \mathbb{R}^m \\ Y &\mapsto y = \tilde{\pi}(Y) \end{aligned} \tag{1.2}$$

To serve our purposes, the out-of-sample extension $\tilde{\pi}$ should be a driving force of our normalization processr. It should respect the modelization of the dataset X given by π in the sense that it provides meaningful low-dimensional counterpart $y = \tilde{\pi}(Y)$ for test subject Y according to the embedding of X; and ideally the obtained representation y of Y should also be the one of its normalized version in high-dimension (the out-of-sample extension should yield the same result as the embedding function on the training dataset, *i.e.* $\tilde{\pi}_X = \pi$).

In addition to finding an out-of-sample extension of our embedding function, we will need to provide a reconstruction function ρ that is able to come back from the low-dimensional space into the high-dimensional one of our sample dataset, as most of the dimension reduction algorithms we used are also not designed from a generative model, and are therefore not provided with an invertible dimension reduction function.

$$\begin{aligned} \rho \colon \mathbb{R}^m &\to \mathbb{R}^d \\ y &\mapsto \rho(y) \end{aligned} \tag{1.3}$$

Just as for the out-of-sample extension, ρ should be as respectful as possible of the model learned by π: applied to the embedding of the dataset X itself, it should provide a reconstruction somehow close to the original image involved, in such a way that after the whole process of projection, "normal" samples should stay relatively unchanged: $\mu_{\mathcal{M}} \approx I_{\mathcal{M}}$. It should also be a part of the normalizing process: images reconstructed by ρ ought to be ones that are part of the control set manifold (and thus present no trace of any anomaly).

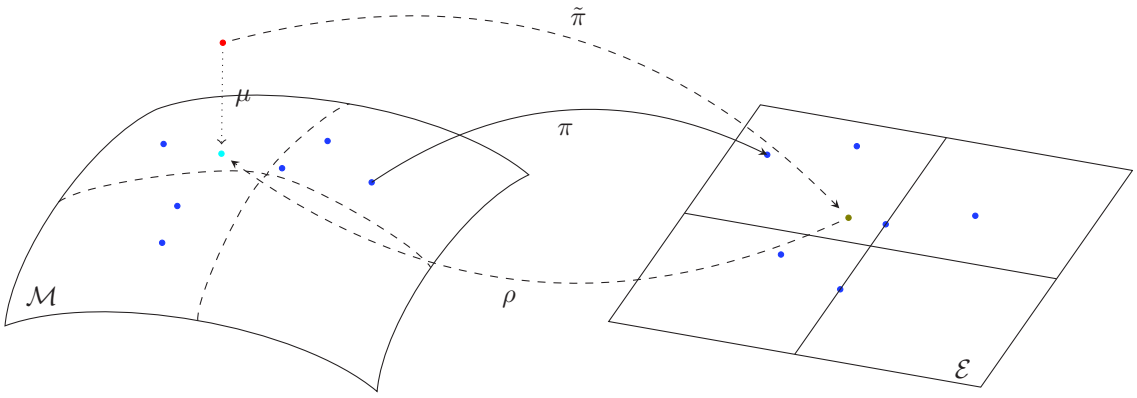

Figure 1.1: Summary of the proposed method. Control samples (in blue) in the manifold $\mathcal{M}$ are reduced in the embedding $\mathcal{E}$ with π. A new sample point (in red) is then embedded in the manifold (olive point) *via* an extension $\tilde{\pi}$. Finally a control correspondent (in cyan) to the test point is found by the reconstruction function ρ, applied to the embedded point.

1.1.3 Anomaly detection

The anomaly detection itself is performed *via* statistical testing. First, we split the training dataset to keep a small portion of control samples aside (that we will call *validation samples*). Once we have learned the model using the training samples, we apply our projection to these validation samples, and subtract these projections to the original images to obtain *residuals* ($R(V) = \mu(V) - V$, with V the *validation set*). A voxel-wise standard deviation σ is then computed over this set of residual in order to statistically confront test samples residuals to this standard deviation: $\forall Y_i \in Y, T(Y_i) = \frac{\mu(Y_i) - Y_i}{\sigma}$, where we denote T our testing function.

1.1.4 The need for robustness

Our proposed paradigm involves projecting a sample over our "normal space", with the idea that the projected sample is a normalized version of our sample. This paradigm is easily performed over normal, or nearly normal samples, but can be much harder for a pathological one. Indeed the heavier an anomaly impacts people, the more it will generate outlier samples (images) that are deeply altered from their healthy versions, and that have a deeply changed relationship towards control samples from the training set.

This is not a problem for univariate methods, as their interest is only in voxel-wise anomaly detection: a pathology affecting a great number of voxels and one only affecting a few will be treated the same. This is of greater concern however for multivariate algorithms such as ours, as they all rely on *some form of proximity* to the manifold: if we recall figure 1.1 it is intuitive that the further a sample is pathological, the further it will be from the manifold

and the harder it will be to get a meaningful global projection of our sample over it.

Hence, for large alterations of our images (a large number of affected voxels, or even a few percentage of voxels that are largely affected), we will need to introduce robust versions of multivariate algorithms as they will inevitably strongly decrease in performance with the percentage of altered voxels. It is noteworthy that our focus for robustness should be on the out-of-sample extension $\tilde{\pi}$, as it is the "normalisation function" of the paradigm: the embedding of a test sample should already have the same characteristic as an control one, and if our out-of-sample extension is robust enough, then the reconstruction does not need to be.

1.1.5 Conclusion

In this chapter, we presented the paradigm of high-dimensional projection that we will use to perform our anomaly detection task. This paradigm is essentially a three-steps problem:

- First, we need to **learn a model over a set of control samples**, and provide a low-dimensional embedding that nicely captures the control samples geometry while untangling a (very) high-dimensional dataset. This step will be performed by **dimension reduction algorithms**.
- Then, we will need to extend this dimension reduction to **all points in the ambient space**, and not just the ones in our control set. To carry out this task, we will introduce **out-of-sample** extensions to our dimension reduction algorithms. We also stressed out the need for multivariate methods to **introduce a robust version** of their out-of-sample extension, capable of dealing with the pathology-induced alteration of our data.
- Finally, we will have to provide **high-dimensional images corresponding to the points in the embedding** as to perform a reconstruction of the low-dimensional samples, thus **completing the projection scheme** we are establishing.

We also presented the final step of our anomaly detection: an individual, voxel-wise, **statistical test of the residual** between our high-dimensional projection and the original test sample. The following chapter will focus on presenting dimension reduction techniques able to cope with the first step of our projection paradigm.

1.2 Dimension Reduction Algorithms

Contents for this section

1.2.1 Linear Dimension Reduction Algorithms

Linear dimension reduction techniques aim to find a linear transformation of our data that provides low dimensional features given high dimensional data points, while trying to maximize a measure of information (*e.g.* mutual information, variance, etc.). We denote X our training data, which lies in $\mathbb{R}^d$, where d is the dimensionality of our data. $\overline{X}$ is the sample mean derived from the training set X ($\overline{X} = \frac{1}{N_s}\sum_{i=1}^{N_s} X_i$), and $\tilde{X} = X - \overline{X}$ the centred dataset.

Principal Component Analysis

Principal Component Analysis (PCA) is the most classic dimension reduction technique. It was originally introduced by Pearson as early as in 1901 [7] and independently developed by Hotelling in 1933 [8]. However, it is truly with the development of computing power and the discovery of the Singular Value Decomposition (SVD) algorithm [9] -a method to identify the singular elements of a matrix- that PCA has grown from a statistical method to a computer science one, which has had a an extremely wide range of applications from chemistry [10] to biology [11].

The aim of the PCA algorithm is to find orthogonal *principal axis* (*i.e.* an orthogonal euclidean subspace of the ambient space) over which data can be linearly projected. The projection over each principal axis (or component) provides a natural dimension reduction for our data: principal components being vectors from an orthogonal basis, the representation of our data's projection in this basis is a low-dimensional vector of dimension the size (cardinality) of our basis. Thus by computing a given number of principal components, we can obtain a representation of our data point of desired dimension.

In practice, the dataset is centred (figure 1.2), and then principal components are computed based on eigenvector decomposition (figure 1.3). Finally, data points are orthogonally projected over these principal components (figure 1.4).

Principal axis can be derived from two equivalent paradigms. The first paradigm is a maximum variance one: principal axis form an orthogonal basis of "directions" (one dimensional vector spaces) which sequentially maximizes the variance of the projected data over each direction. The first principal component is the unit vector such that the data projection over this vector is

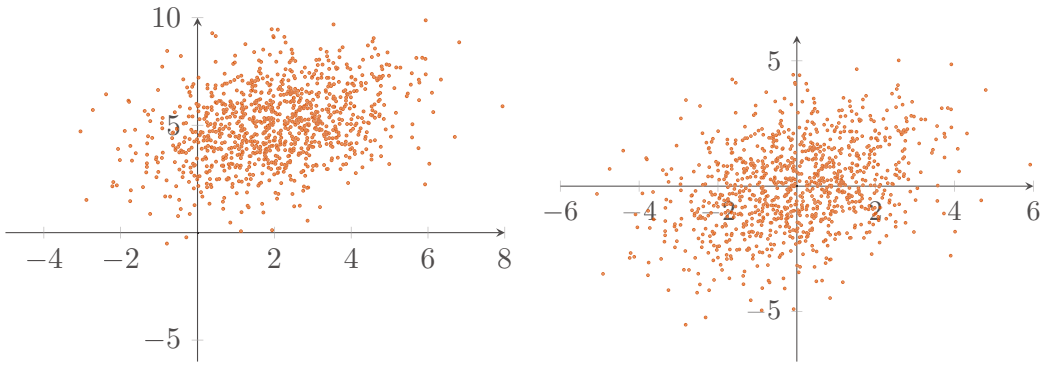

Figure 1.2: PCA algorithm: Left panel presents the original dataset (in orange), a two-dimensional multivariate heterostedastic normal distribution. Right panel showcases the dataset after being centred.

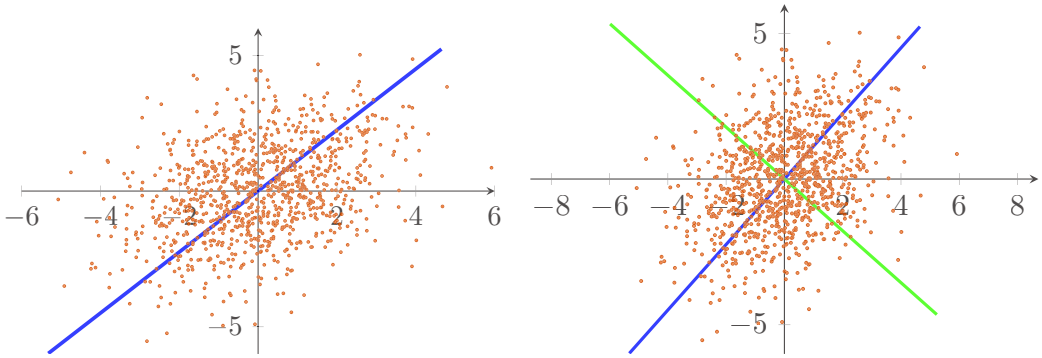

Figure 1.3: PCA algorithm: After centring the dataset, the first principal component is derived from the covariance matrix eigenvectors. The direction associated by this component is given by the blue line, while the direction given by the second component is given by the green line in the second panel.

of maximal variance; the second principal component is then the unit vector that maximizes the projected data variance along every direction orthogonal to the first principal component (*i.e.* the PCA algorithm can be viewed as a recursive algorithm finding the $p+1$-th principal component over the residual of the original data and its orthogonal projection over the set of the first p ones in a Gram-Schmidt fashion).

Let us denote $(u_1, \ldots, u_m)$ the m first principal components of the dataset X. The first paradigm to find the first principal component of X can be written as follows:

$$u_1 = \arg\max_{u \in \mathbb{R}^d} \mathrm{var}(X.u) \text{ s.t. } \|u\|_2 = 1 \tag{1.4}$$

and any principal component u_{p+1} can be derived given the first p principal components in the following fashion:

$$u_{p+1} = \arg\max_{u \in \mathbb{R}^d} \mathrm{var}(R_p.u) \text{ s.t. } \|u\|_2 = 1,\ u \perp (u_1, \ldots, u_p) \tag{1.5}$$

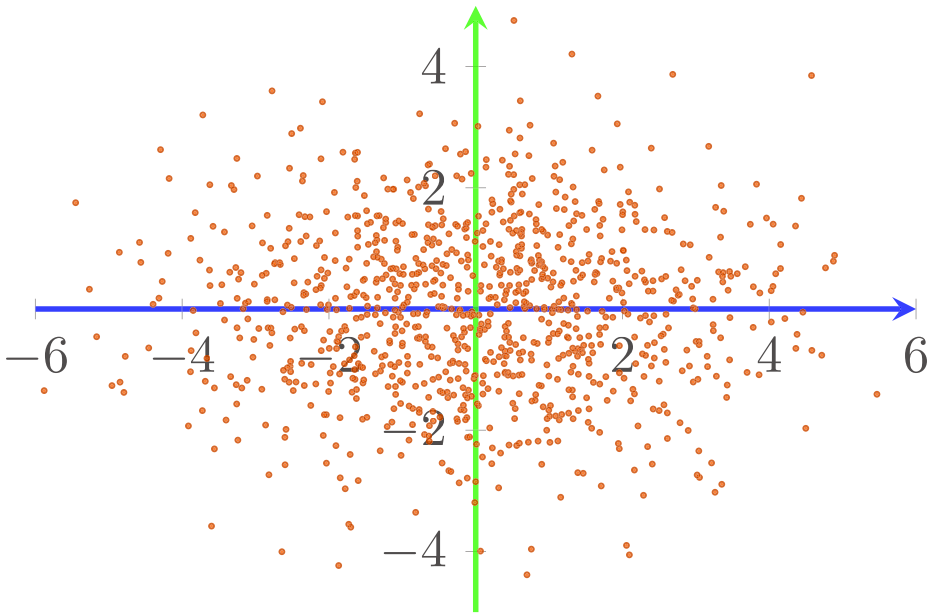

Figure 1.4: PCA algorithm: Finally, the data is projected in the principal component space. As we computed as much principal components as the ambient space dimension, PCA here is just a change of frame (specifically a rotation).

where R_p is the residual (of order p) between X and the projection of X over the p first principal components, *i.e.* for each sample X_n in X, we have

$$X_n = \sum_{i=1}^{p} < \tilde{X_n}, u_i > u_i + R_{p,n} + \overline{X}$$

With $R_p \in \mathbb{R}^{N_s \times d}$, and $R_{p,n} \in \mathbb{R}^d$.

The projection model for PCA can then be written for each sample X_n as: $\tilde{X_n} = \sum_{i=1}^{m} u_i x_{n,i}$, with $\forall i \in [\![1, N_s]\!]$ $x_{n,i} =< \tilde{X_n}, u_i >$ $(x_n \in \mathbb{R}^m)$.

This can be rewritten for convenience:

$$X_n = W.x_n + \overline{X} \tag{1.6}$$

$$x_n = W^T(X_n - \overline{X}) \tag{1.7}$$

Where

$$W = \left(\begin{array}{c|c|c} u_1 & \ldots & u_m \end{array} \right)$$

is the matrix of principal components called the projection matrix.

The second paradigm (which, as we previously mentioned, has exactly the same solution as the first one), is a reconstruction one. Principal components form the orthogonal subspace that best embeds the data in a compressed sensing view: that is to say the projection over the principal components is the linear orthogonal projection that, after reconstruction, is the closest to our

original data with a L_2 criteria. As we are looking for a number of components far lesser than the original dimension of our data, the "compression" notion associated with PCA appears quite intuitively with this paradigm: we can learn the reconstruction model, and only store the samll sized dimension reductions of our samples.

With the same notations as before, the second paradigm can quite easily be formulated as finding the orthonormal basis $u = (u_1, \dots, u_p)$

$$u = \arg \min_{u \in \mathbb{R}^{p \times d}} \|R_p\|_2 \tag{1.8}$$

It can be shown that principal components are actually the eigenvectors of our training dataset covariance matrix C, associated with the largest eigenvalues:

$$C = \frac{1}{N_s} \sum_{i=1}^{N_s} (X_i - \overline{X})(X_i - \overline{X})^T = \frac{1}{N_s} \tilde{X} \tilde{X}^T$$

It can also easily be shown that, as proposed in the first paradigm, each eigenvalue associated with a principal component corresponds to the variance of our data in the direction of the principal component, *e.g.* the direction of largest variance in X is given by u_1, and the associated variance is λ_1, where $(\lambda_1, \dots, \lambda_{N_s})$ are the eigenvalues of C. Finding the eigen elements of C being closely linked to finding the singular elements of X, the SVD algorithm is thus particularly helpful in applying PCA to a dataset. One can derive from this a very useful property to "automatically" select the number of components that ought to be kept (or respectively at which we can stop), which is the proportion of explained variance:

$$C_p = \frac{\sum_{i=1}^{p} \lambda_i}{\sum_i \lambda_i} \tag{1.9}$$

this ratio is a guide on how much of the data has already been "explained" by the first p components of PCA. A classic threshold for the proportion of explained variance is the infamous 0.9 mark.

As we have seen in eqs. (1.6) and (1.7), the solution of both paradigm as a dimension reduction for the training set X has the form of $x_n = (< \tilde{X}_n, u_1 > , \dots, < \tilde{X}_n, u_1 >)$, and the associated high-dimensional projected point $W x_n + \overline{X}$. Furthermore, it can be shown that the out-of sample projection of any given sample Y (not necessarily in the training set X), if written as such, is also the solution to the least-squares problem for projecting Y over X principal directions of variance. Therefore, we can now connect the PCA algorithm to the projection paradigm we introduced in section 1.1.2. The embedding of our data point Y in the low-dimensional euclidean space $\mathcal{M}$ is given by $y = \tilde{\pi}(Y) = W^T(Y - \overline{X})$, its reconstruction $\rho(y) = Wy + \tilde{X}$, and the associated high-dimensional projection is then just $\mu(Y) = \rho \circ \tilde{\pi}(Y) = WW^T(Y - \overline{X}) + \overline{X}$.

Figure 1.5 presents the projection of a set of noisy, linearly distributed, data points, such that the projection of each two dimensional data point is

a real-valued one dimensional feature, representing the dot product between our data point and the principal vector obtained by the PCA, which is a direction vector of the orange line. Figure 1.6 showcases another test in which the data is this time non-linearly distributed on a one dimensional manifold that is voluntarily approximated for example purposes as a one dimensional euclidean space by the PCA algorithm. We can see that in this case, both the low-dimensional reduction and high-dimensional projection are untrue to the data distribution, as points diametrically opposed on the circle have nearly the same projection and reconstruction. Thus we will have to carefully select the number of components that we will use while performing linear dimension reduction.

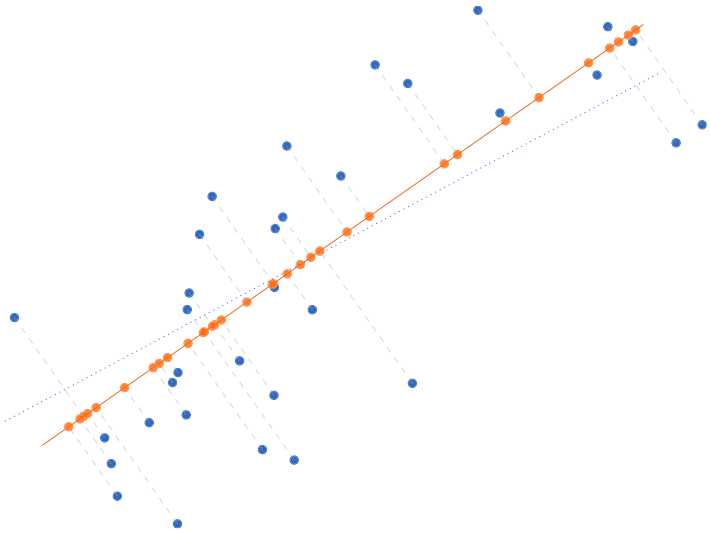

Figure 1.5: Projection of samples over the first PCA component in a regression scheme: the blue dotted line represents the original model from which data points are sampled before adding random gaussian noise. The orange line represents the linear regression corresponding to the L_2 minimization presented in the second paradigm of PCA. Dotted line between the orthogonal projection (orange points) and original points corresponds to the residual.

Remark: As it is based on an eigen decomposition, PCA (and all eigen-based dimension reduction algorithms for that matter) is limited by the fundamental theorems of linear algebra. In this context, by trying to decompose a matrix of size $N_s \times d$, we cannot hope to recover more eigenvectors than $\min(n_s, d)$ (and more generally no more than $\text{rank}(X) \leq \min(n_s, d)$). As the number of dimensions we have to deal with is *generally* far greater than the number of samples we can rely on, the number of principal components that we are effectively able to compute is lesser than the number of samples we gathered.

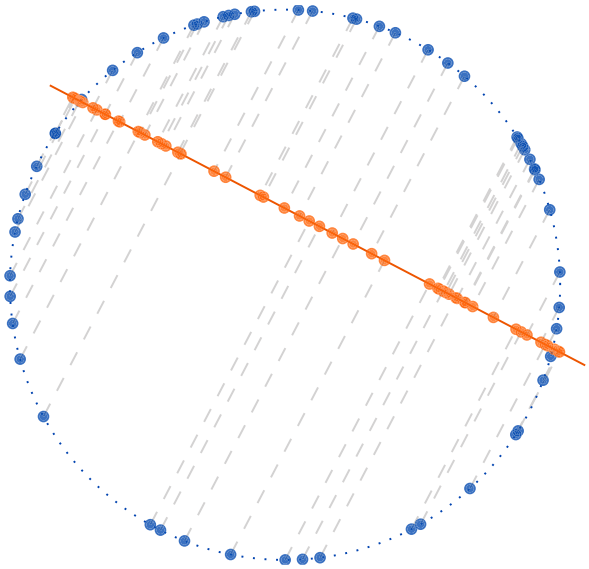

Figure 1.6: Projection where the data is non linearly distributed. Here data points are sampled over a circle (a one dimensional manifold), and we project them over a one dimensional euclidean space in a linear fashion with PCA. The obtained projection is widely inadequate.

1.2.2 Non Linear Dimension Reduction Algorithms

Ideas for non-linear dimension reduction techniques have been developed as early as in the 1950s with noticeable examples such as Multidimensional Scaling (MDS) [12, 13], whose purpose is to find points in a lower dimensional space that best respects a certain metric between points in the high-dimensional space (usually euclidean distances, or rank ordering of these distances), or later the Self Organizing Map (SOM) algorithm [14, 15] that fits a grid in low dimension to high-dimensional data to find a non-linear mapping from a low-dimensional space to the data. The kernel PCA algorithm (KPCA) [16, 17], provides a kernel extension to the previously introduced PCA algorithm that, by using the kernel trick [18], is also able to project non linearly distributed data along greatest variance directions.

The emphasis of all these techniques was always to modelize a set sampling a manifold embedded in an euclidean space, albeit a (very) low-dimensional one. Our focus in this thesis was on more recent non linear methods inspired or directly derived from these ones.

The interest of non-linear dimension reduction methods is obviously highly dependent of the dataset it is aimed to be used on. In the case of samples distributed as in figure 1.5, none of them would be more relevant than PCA for instance, as this dataset is linearly distributed (with a bit of noise). However, it is quite clear from figure 1.6 that there is a lot of improvement to be expected from non linear methods over such a kind of dataset, provided that the non linear algorithms we use are able to “untangle” the non linearities in the data

with the correct amount of dimensions.

Isometric Feature Mapping

Isometric Feature Mapping [19, 20] (Isomap) is one of the currently most used non-linear dimension reduction algorithm. As the vast majority of other non-linear algorithms, it relies on the hypothesis that our data lies in a low-dimensional manifold, set in a high dimensional space, and it is directly relying on this assumption in the core of its algorithm. Indeed, Isomap aims at providing an embedding for a training set in an *isometric* way, that reproduces its global geometry by preserving in the low-dimensional space an estimation of geodesic distances (geodesic distances are the closest distances between two points on a manifold, while this distance must be the length of a curve linking the two points by staying in the manifold, see figure 1.7) between the dataset samples. Such an approach (isometric), is inherently flawed to deal with non isometric manifolds [20, 21] (manifolds that are isometric result from an isometric mapping of a closed, convex set in a low-dimensional euclidean space, into a high-dimensional one), as we will see further on, but still very effective on natural image manifolds.

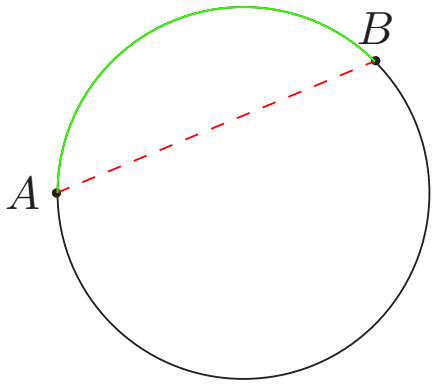

Figure 1.7: A trivial illustration of the geodesic distances between two points over a manifold. Here the manifold is a one dimensional circle manifold embedded in the two dimensional plan. The red path links both points by realizing the euclidean distance, while the green curve links the points by staying on the manifold (and is of minimal length). Its length is the geodesic distance between A and B.

In the case of Isomap, said estimation of geodesic distances is done by creating a graph over our training samples. This graph is created by performing a k-neighbourhood search (with L_2 euclidean distances to find neighbours) over our samples, and computing the associated distances between them. Then, with the help of Dijkstra's algorithm [22] as a shortest-path finding algorithm,

we approximate pairwise geodesic paths between our samples by a set of local neighbourhood-constrained paths, ensuring that a global path links two samples by "travelling" from sample to sample while each local step is done in the neighbourhood of the current sample (see figure 1.8 for a visualisation of the distance computation).

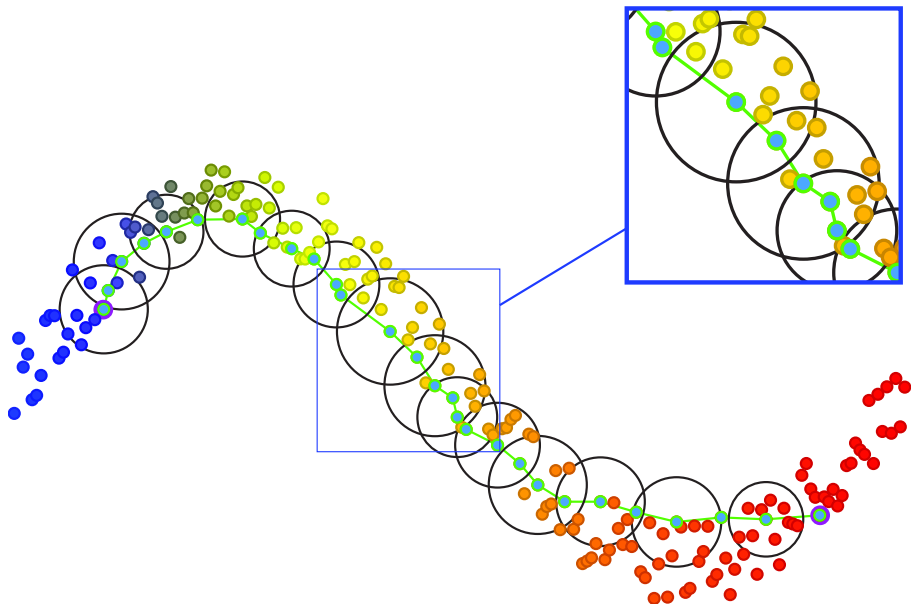

Figure 1.8: A showcase for the geodesic distance computation by the Isomap algorithm. Circles represent the neighbourhood of the data points on the path (only half of them are represented). The global path (in green), links the starting points (purple fringed points), by assembling local paths (in green) that pass from neighbourhood to neighbourhood. Dijkstra's algorithm ensures that given such constraints, the global green path is the shortest one between both starting points.

Providing a low-dimensional embedding is done while attempting to preserve the distances by ways of classical Multidimensional Scaling [23] (MDS). Essentially what the classical MDS does, given pairwise distances between all of our sample points and a lower dimension m, is trying to set points in the lower dimensional space (or embedding) corresponding to the ones in our training data while keeping distances between embedding points as close as possible as the precomputed ones. This amounts to finding the solution of the following problem:

$$(x_1, \dots, x_{N_s}) = \arg \min_{\mathbb{R}^{N_s \times m}} \frac{\sum_{i,j} (d_{ij} - \hat{d}_{ij})^2}{\sum_{i,j} d_{ij}^2} \tag{1.10}$$

with $\hat{d}_{ij} = \|x_i - x_j\|^2$ being the euclidean distances between two sample dimension reductions (*i.e.* distances in feature space) and d_{ij} the original distances. Interestingly, the solution to this problem is quite close to the PCA solution previously introduced, in the sense that it is also provided by the eigenvectors

of an affinity matrix. However this time, instead of a covariance matrix, MDS provides one created from the geodesic distances that has been normalized by double centring: $K' = -\frac{1}{2}JKJ$, where $J = I_{N_s} - \frac{1}{N_s}\mathbb{1}\mathbb{1}^T$. The double centring is a way to transform a distance kernel into a dot-product one.

Locally Linear Embedding

Locally Linear Embedding (LLE) has been introduced [24] nearly simultaneously to Isomap. The LLE algorithm has obviously the same purpose as Isomap (performing a non-linear dimension reduction of a training dataset that "untangles" the data in the embedding) and both algorithms are using the manifold hypothesis (and thus manifold properties) to achieve this purpose. However, the two algorithms differ in the way they apply these manifold properties to obtain their embedding.

As the name suggests, LLE is a locally linear method. It makes use of the local linear property of the manifolds (see appendix 5.1.3), whereas Isomap was relying on the existence of a distance intrinsic to the manifold, (supposedly) more suited to express relationships between data samples. Here, LLE is applying a linear algorithm of dimension reduction in neighbourhoods of each data point where, under correct sampling of the manifold and given a correct choice of embedding dimension, the dataset is indeed *locally* linear.

For each sample, LLE provides a linear projection of the data point: (linear) barycentre weights according to its closest L_2 neighbours are computed, and the algorithm then tries to find a low dimensional space that ensures all these barycentric weights are respected, in the same way that Isomap would attempt to respect ordering in pairwise distances.

The LLE optimization problem of finding the weights for each neighbours of each sample given in equation 1.11 can analytically be solved [24, 25].

$$\begin{aligned} \forall X_n \in X,\ W_p = \arg\min_w \|X_n - \sum_{i\in\mathcal{N}(X_n)} X_i w_i\|_2, \\ \text{s.t. } \sum_i w_i = 1 \end{aligned} \tag{1.11}$$

where $\mathcal{N}(X_n)$ is the neighbourhood of our sample X_n (*i.e.* the indices of training samples closest to it in euclidean distance). Finding a corresponding embedding that best preserves the weights computed for each sample is a matter of optimizing the following quadratic form:

$$(x_1, \ldots, x_{N_s}) = \arg\min_{x\in\mathbb{R}^{N_s\times m}} \sum_i \|x_i - \sum_j (W_i)_j x_j\|^2 \tag{1.12}$$

which, as a quadratic form, is coincidentally solved *via* an eigenvector decomposition, with the following constraints to make it a well-posed problem: $\sum_i x_i = 0$ (centred embedding) and $\frac{1}{N_s}\sum_i x_i x_i^T = I_{N_s}$ (unit covariance embedding).

An illustration of the algorithms two steps is given in figure 1.9, and an execution of the algorithm over three toy example datasets will be provided in figs. 1.10, 1.11 and 1.13.

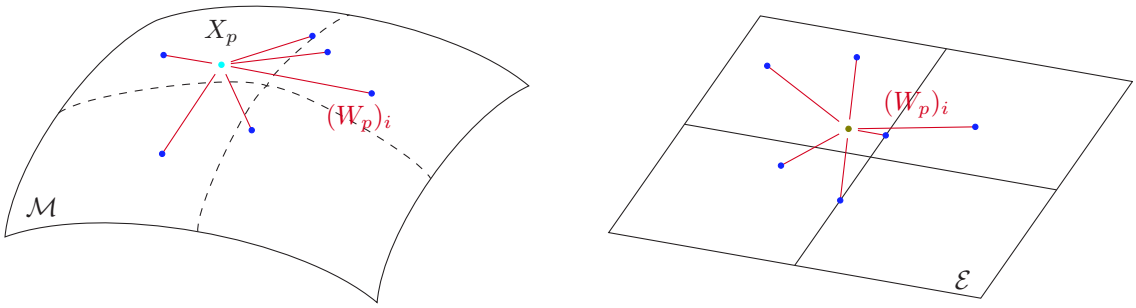

Figure 1.9: An illustration of the LLE algorithm. Left panel presents the high- dimensional training data in its underlying manifold. Barycentric weights $(W_n)_i$ are learned for each sample X_n in an analytic fashion. The sample data is then projected in a low-dimensional embedding (right panel) that ensures that barycentric weights are respected for each low-dimensional sample. The embedding supposedly captures the data inherent non-linearities to restore a linear behaviour in low-dimension.

Diffusion Maps

Diffusion Maps have been developed since 2005 [26, 27, 28] and are a generalization of numerous non-linear dimension reduction techniques (laplacian eigenmaps [29], hessian eigenmaps [30], or even LLE) that uses kernel and graph properties such as deriving the laplacian of a graph [31, 32, 33] (*i.e.* the difference between the matrix of its vertices degrees and its adjacency matrix), to perform their dimension reduction. Diffusion Maps aim at providing a good geometric representation of our dataset by using a different paradigm of dimension reduction than the ones used by PCA or Isomap.

The main idea here is to represent our data on a connected graph. We then define a Markov chain over the data graph by designing a transition matrix between our different data samples. The Markov chain is designed by using a kernel over the usual pairwise L_2 distances between samples. The kernel that will be used (usually the gaussian kernel, as it has the nice property of being invariant by rotations and translations) is an approximation of the "local geometry in X" and each kernel, having its inherent properties, can have a different effect over the dataset modelization. The newly designed Markov chain over our dataset is the direct modelization of a heat diffusion process from which the algorithm gets its name. Using a kernelized version of the euclidean distances yields as a result a new distance called the diffusion distance, that somehow is an approximation of the degree of difficulty to reach a point starting from another one over the manifold $\mathcal{M}$ (or the mean time needed to travel

between both point in a sense). One selling point of the Isomap and LLE algorithms was their obvious robustness to many geometric transformations of the training data: neighbours weights are unaffected by rotations, translations or scalings (homothetic transformations). This diffusion distance, while only being to one the two former (rotations), offers the advantage of being more robust to outlier [26, 34] and to be differentiable (which will be particularly helpful in optimization schemes).

A natural dimension reduction arises when we consider that we modelized our dataset with a Markov chain: the behaviour of the chain over long periods of transitions is a natural way of finding out the profound relationships between samples (samples that are frequently "visited" together by the chain over a short period of time are bound to be part of the same geometrical area in $\mathcal{M}$). Recalling that the long-term behaviour of a Markov chain is given by its first eigenvectors, we get intuitively that the local geometrical aspect of our manifold can be captured by the eigenvectors of the normalized laplacian corresponding to the largest eigenvectors of L. A quick pseudocode for the Diffusion Maps algorithm is given in algorithm 1.

Algorithm 1 Diffusion Maps algorithm

Require: X dataset, γ

$d_{ij} = \|X_i - X_j\|^2$ ▷ Compute pairwise distances
$K_{ij} = \exp(\frac{-d_{ij}}{2\gamma})$ ▷ Compute affinity matrix
$d_i = \sum_j K_{ij}$
$K_{ij} = \frac{K_{ij}}{\sqrt{d_i dj}}$ ▷ Normalize the kernel matrix
$d_i = \sum_j K_{ij}$
$L_{ij} = \frac{K_{ij}}{d_i}$ ▷ Normalize the Laplacian matrix
Return eigenvectors of L

1.2.3 Toy Examples

We designed several toy examples to highlight the different strengths and flaws of the dimension reduction techniques presented in the previous section. The two first are two dimensional manifolds embedded in the three dimensional space, but with different properties. The first is the "fish bowl" dataset: a severed sphere uniformly sampled from two angles (polar and azimuthal angles) but for which the elevation (or altitude) is limited to a range strictly included in the one of the original sphere (for instance an altitude inferior to 0.8 for a unit sphere). Parametrization for the fish bowl according to its two angles is

given in equation 1.13.

$$\phi \in [0, 2\pi], \theta \in \arccos([-1, 0.8]), \begin{cases} x = \sin(\theta)\cos(\phi) \\ y = \sin(\theta)\sin(\phi) \\ z = \cos(\theta) \end{cases} \tag{1.13}$$

The second classic dataset is the "swiss roll" dataset, which is basically a rectangular layer that has been rolled over itself in a spiral fashion and is thus embedded in the three dimensional space. The swiss roll parametrization according to the original width of the rectangle and to the angle resulting to the curvature is given in equation 1.14.

$$\phi \in [0, 3 * \pi], Z \in [0, 10], \begin{cases} x = \sin(\phi)\frac{\phi}{6} \\ y = \cos(\phi)\frac{\phi}{6} \\ z = Z \end{cases} \tag{1.14}$$

Lastly, a more practical toy example dataset is given by the "astronaut" dataset, for which we naturally embed a two-dimensional manifold into a high-dimensional space: indeed we consider an original image (the astronaut image of figure 1.12) over which we apply a set of two different transformations (whose parameters are uniformly sampled over a rectangle), a rotation and a scaling. The rotation angle is sampled between 0 and π, while the images can be scaled from 10% of the original size (with zero padding) to full size. By composing both transformations over the original image, we create a dataset in the original image space of very high dimension (256×256), linked to our 2D rectangle by an (obviously) non-linear (and non-trivial) function.

Figures 1.10, 1.11 and 1.13 present applications of the PCA, Isomap, LLE and Diffusion Maps algorithms over our three toy datasets.

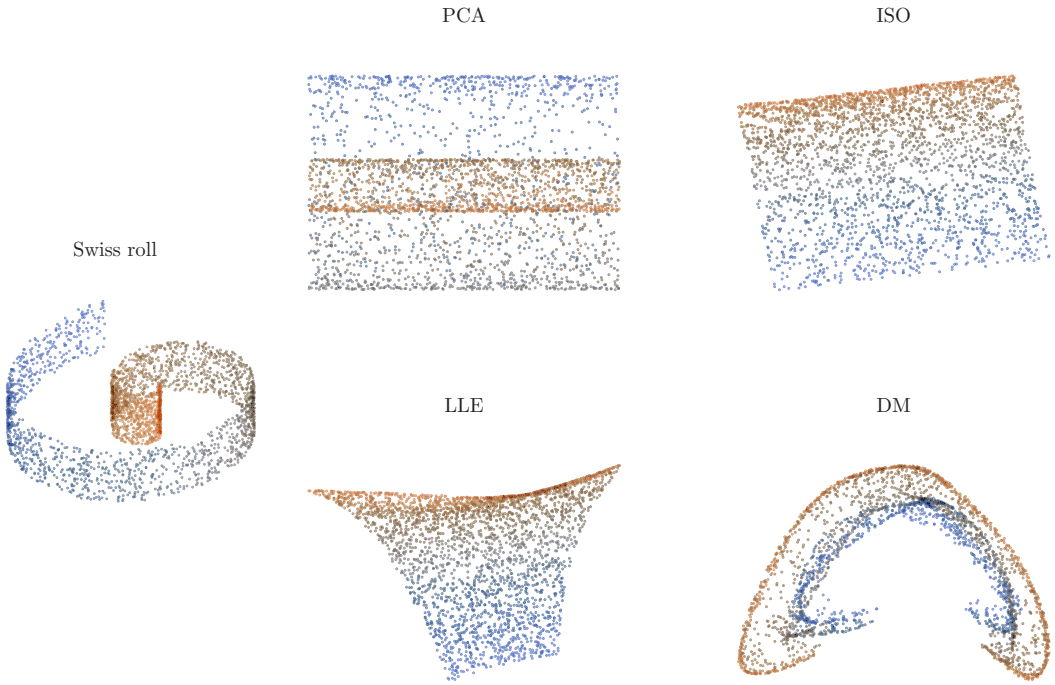

Figure 1.10: A classic toy example over which we applied our algorithm. Left panel presents the original 3D dataset that is sampled on the "swiss roll" 2D manifold, and the four right panels showcase the 2D embedding obtained from our dimensions reduction algorithms on this training set.

As we can see from figure 1.10, the PCA algorithm is unable to provide an untangled reduction dimension for the swiss roll dataset and provides merely an orthogonal projection over two of the three basis vectors, while Isomap is perfectly able to deal with the swiss roll, and its inherent data geometry is nicely recovered. The application of the LLE algorithm over the swiss roll is nearly as good as the one from Isomap: even if the inherent geometry of the dataset is not correctly estimated by LLE, the embedding does manage to untangle the manifold from its three dimensional structure to a correct two dimensional one, in which the relationships between our samples are linear. The dimension reduction of the diffusion maps, while being the most peculiar is nearly correct: even if the behaviour in this low dimensional subspace cannot be linear, local neighbourhoods of samples are constituted according to their sampling parameter, and the swiss roll is partly untangled.

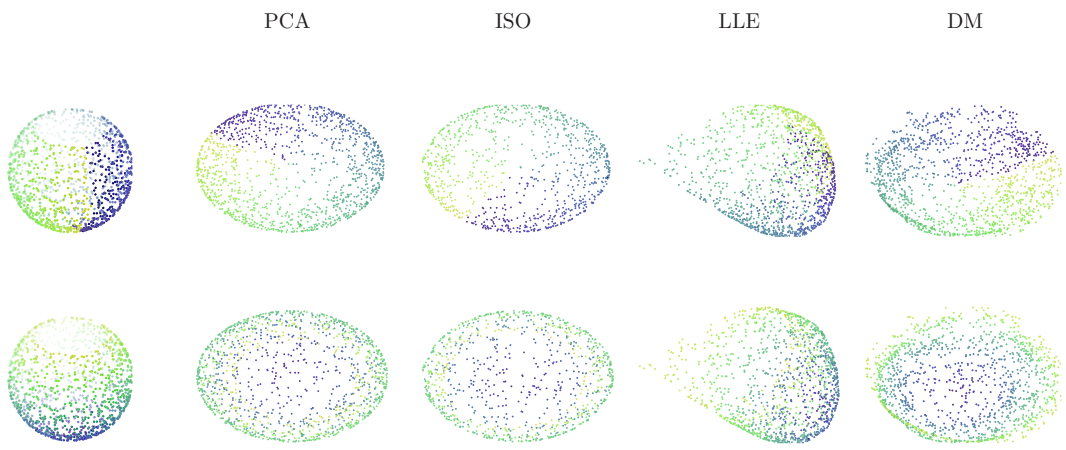

Figure 1.11: A classic toy example over which we applied our algorithm. Top left panel presents the original 3D dataset that is sampled on the "fish bowl" 2D manifold, and the four right panels showcase the 2D embedding obtained from our dimensions reduction algorithms on this training set. Top row is coloured with a colormap indexed on the azimuthal angle ϕ, and bottom row is coloured with a colormap indexed on the polar angle θ

The fish bowl dataset is absolutely not correctly embedded, neither by PCA, Isomap nor LLE (although the embedding provided by LLE is better than the Isomap one), and this is due to the fact there is no possiblity that a global method such as Isomap (or LLE), with its underlying isometric assumption, would find a way to represent the fish bowl in two dimensions while correctly representing the geodesic distances computed over the severed sphere in two dimensions (it would violate the fact that the sphere is indeed a non-euclidean manifold and cannot be mapped isometrically from a set of $\mathbb{R}^2$ into $\mathbb{R}^3$). Here the dimension reduction of Isomap is merely a projection in the most basic sense over the xy plane, while the dimension reduction of LLE slightly untangles the fish bowl but is not completely able to separate points that are nearly diametrically opposed on the sphere. The dimension reduction obtained by the Diffusion Maps on the fish bowl dataset is quite satisfying: the two dimensional embedding nicely "opens" the sphere in the most intuitive way to obtain a nearly optimal embedding for this dataset.

Figure 1.12: The astronaut image (from astronaut Eileen Collins, courtesy of NASA Great Images database) which originates the eponym dataset.

The results of our algorithms on the astronaut dataset are displayed on figure 1.13. Each scatter plot is coloured according to the rotation parameter used to sample our points. We also selected six representative samples from the dataset that we pointed out in each scatter plot. As before, the two dimensional representation found by PCA is not a good one for our dataset, as it mixes points with disparate inherent parameters of rotation and scaling in very close neighbourhoods. The Isomap and LLE embeddings are both excellent, untangling the data in the two dimensional plane in which we have a linear behaviour, and respecting both sampling parameters (as can be observed by the colormap and the six representative images repartition). The Diffusion Maps result is ambiguous: we do obtain an untangled representation of our dataset in two dimensions, also respecting both parameters, but the relationship between our points inside this representation is noticeably non linear according to the scale parameter: a small fraction of the distance required to get from the lowest scaling to full scaling is sufficient to get to half of the full scaling parameter. This can affect algorithms of reconstruction based on distances for instance.

It is crucial to denote that, while being hopelessly inefficient to recover the true geometric structure underlying a manifold embedded into a higher dimensional space given the correct number of dimensions, PCA is perfectly able to do so using a higher one: figure 1.14 presents two three dimensional views of the embedding obtained by PCA with its three first components. It is noticeable that now it is a much more correct dimension reduction for our dataset. Thus the only worry we should have while using PCA as a dimension reduction algorithm, even on non linear datasets, is making sure we use enough components in our dimension reduction.

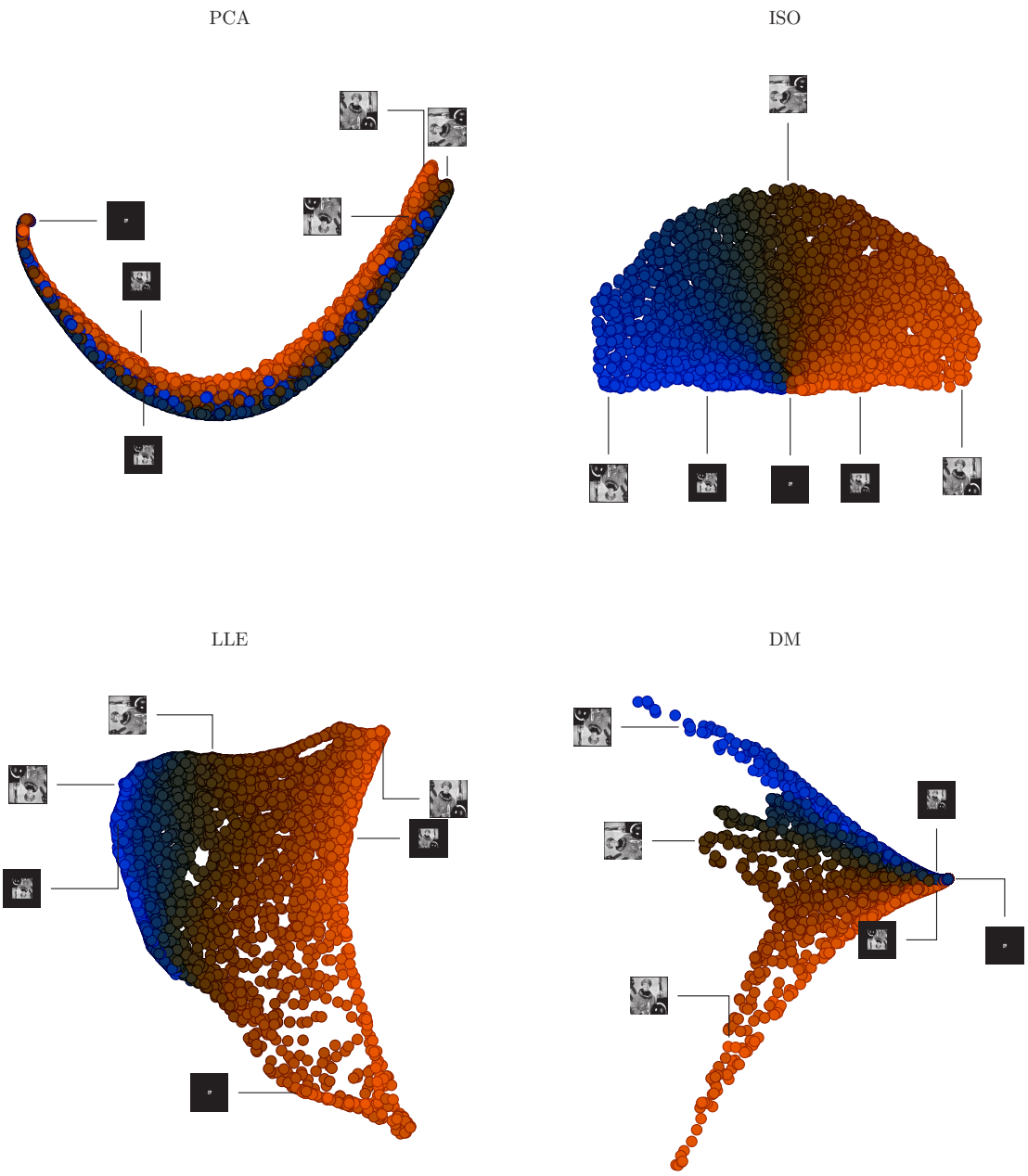

Figure 1.13: A classic toy example over which we applied our algorithm. The four panels represent the dimension reduction of each of our four algorithms over the astronaut dataset. Colormap is done accordingly to the rotation angle (as it is the hardest to recover). The same six examples have been pointed out in each algorithm and are meant to be representative of the dataset.

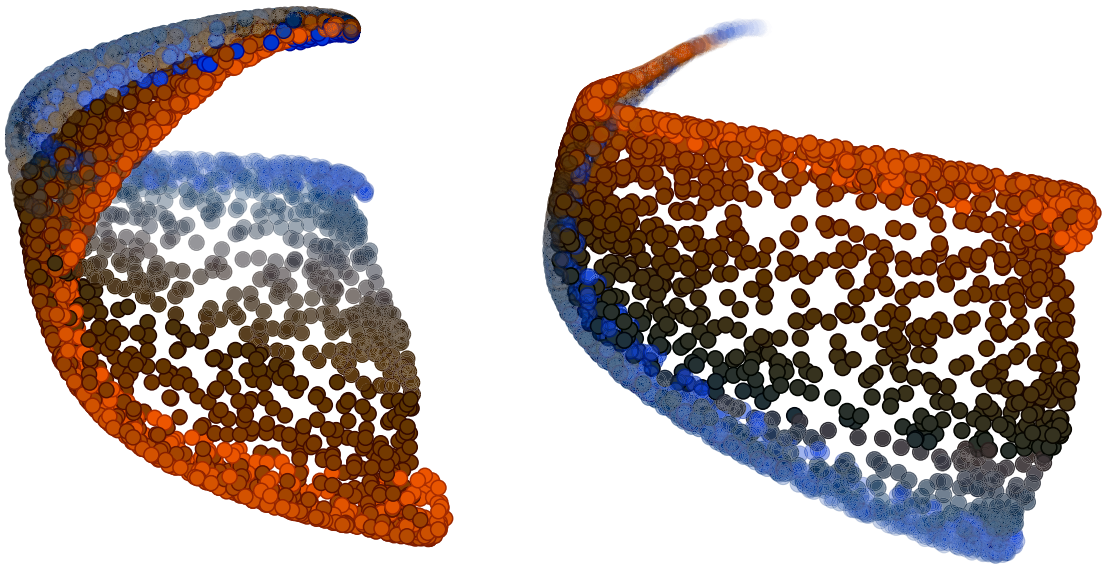

Figure 1.14: Three dimensional views of the astronaut dataset dimension reduction obtained by PCA. The embedding of PCA in 3D spans a two dimensional manifold.

1.2.4 Conclusion

We have presented a wide variety of dimension reduction algorithms as our first step of anomaly detection. For linear dimension reduction algorithms we focused on the **Principle Component Analysis** algorithm, while we presented three non-linear algorithms: **Isomap, LLE** and the **Diffusion Maps**. We also presented a small set of toy examples over which we applied our algorithms, allowing us to gather some preliminary results about the behaviour of these techniques, which can be synthesized as follows:

- **Non-linear dataset are impossible to be recovered with the correct amount of dimension with linear methods**, and can even be **very challenging for the non-linear ones** depending on the geometric structure and the sampling of the underlying manifold.
- Given **enough dimensions**, PCA can find an acceptable dimension reduction for a non-linear dataset.
- The number of dimensions that we will use to represent our dataset is **limited by the number of training samples** available.
- All methods of dimension reduction show **impressive results** in reducing the dimensionality of **image datasets**.

While we solved the problem of getting a low-dimensional representation of our training data that untangled it while respecting its inherent geometry,

we did not provide (apart from the PCA algorithm) a way of applying this reduction dimension to any sample that we wish to test for anomaly detection. In the following section, we will strive to provide an *out-of-sample* extension for the non-linear dimension reduction algorithms.

1.3 Out-of-sample extension

Contents for this section

As we previously mentioned in sections 1.1 and 1.2, a major downside of *most* dimension reduction algorithms is that once an embedding has been provided for the training set and a model of normality has been learned, the projection of a new sample is not as straightforward as one could expect, as these techniques were mostly designed to be used only once over a whole dataset and not several times over different datasets.

In this section we will address the problem of providing an out-of-sample extension to the dimension reduction methods we presented in 1.2 and obtain a model that we can easily apply to any given images of the space from which the control samples are sampled (or any given vector of the ambient space for that matter). This is not an easy problem as the samples for which we want to find an extension do not necessarily belong to the manifold of "normal" subjects $\mathcal{M}$ and that as an additional constraint, we wish for the extension of any point not belonging to $\mathcal{M}$ to have an extension close to the one that its unaltered counterpart would have.

We would also wish to ensure, as stated in section 1.1.4, that our out-of-sample extension is somehow robust. Indeed, the dimension reduction of the training set itself has no need for robustness (although it *could* be useful in the case of a training set polluted with outliers) as it is only dealing with subject which are representatives of the normality and thus present no sign of a pathology. The second step of our paradigm being the out-of-sample extension applied to test samples, this is our first and main opportunity to introduce robustness in our algorithms, as to avoid for pathological samples to have an accordingly abnormal dimension reduction (if test samples have the dimension reduction corresponding to their healthy counterparts, then there is no need for a robust reconstruction).

We will first take a look at simple case of the PCA algorithm.

1.3.1 PCA out-of-sample extension

As we already expressed in 1.2.1, PCA provides a natural out-of-sample extension that is analytically derived from the dimension reduction function obtained in the training phase over the control samples: $\pi(X) = W^T(X - \overline{X})$ and $\tilde{\pi}(Y) = W^T(Y - \overline{X})$. We also stated in 1.2.1 that with such a definition, the out-of-sample extension for Y is also the orthogonal projection of Y over the principal components of X. This gives PCA the unique property of having both an analytical expression for the dimension reduction (as opposed to non-linear dimension reduction that only provide the final low-dimensional

points), and the out-of-sample extension. Unfortunately, non-linear dimension reduction algorithms do not share this extremely helpful property and we will need to cope for it ourselves to provide an out-of-sample extension hopefully as equally endowed with statistical properties as the PCA one.

The PCA out-of-sample extension is not inherently robust in our context of anomaly detection, as it heavily relies on L_2 distances or relationships, and as this distances can be largely affected by the pathology we are focusing on. Therefore, in an effort to make it more robust to anomalies, Vik et al. [1, 35] have proposed a robust PCA-based algorithm for data projection in high dimension (*i.e.* the hereby presented paradigm) able to cope with a pathology altering test samples. An additional modelization [35] can be made in the embedding space by adding a prior distribution (mixture of gaussians or non-parametric) to the training set dimension reduction, thus transforming the robust, maximum likelihood estimation into a maximum *a posteriori* estimation, which in turn can be solved by adding a mean shift procedure to the M-estimation scheme.

Let us now look into the extension of non-linear dimension reduction techniques.

1.3.2 The Nyström extension

Fortunately for non-linear methods, we can make use of an old eigenfunction extension technique called the Nyström extension [36, 37], that was originally designed to help finding numerical solutions to integral equations. It has been widely used over the recent years in large dataset sampling, to approximate a kernel with fewer data points in order to reduce memory and computational costs [38, 39]. In our context, we will use the Nyström method to provide an out-of-sample extension to our projection method π. This extension is two-fold: as it is a kernel method, we first need to extend the kernel K used in our non-linear dimension reduction algorithms (we will denote $\tilde{K}$ this extension). The extended kernel is then used to provide the out-of-sample extension given the Nyström extension formula. Most of the kernel extensions for dimension reduction algorithms can be found in [40]. Each kernel extension is designed to be consistent with the natural properties of the kernel that is being used (geodesic extension of the Dijkstra's algorithm used in Isomap when adding a new data point to the training graph, local neighbourhood extension for LLE and the diffusion process of the Diffusion Maps). The idea being that we are using a kernel that is defined over our whole ambient space, but only computed this kernel over our training set. The kernel extensions for Isomap, LLE and the Diffusion Maps can be found in equations eqs. (1.15) to (1.17).

$$\tilde{K}_{ISO}(Y, X_i) = \min_{U \in \mathcal{N}_X(Y)} K_{ISO}(U, X_i) + d(Y, U) \tag{1.15}$$

From equation 1.15, we get that the Isomap kernel extension is fairly intuitive: a new point kernel extension is based on the one of its neighbours, as it is done

for any point in the training set. As we are computing geodesic distances, the geodesic distance from the new point to any point in the training set is derived as the path of minimal distance from it through one of its neighbour to the destination.

$$\tilde{K}_{LLE}(Y, X_i) = w(Y, X_i) \tag{1.16}$$

The extension 1.16 of the LLE kernel is straightforward. The LLE kernel is neither a distance nor a dot-product kernel, but a (barycentric) weight kernel. The extension to a new point is simply the computation of the barycentric weight of this test point Y over the dataset X: $w(Y, X_i) \neq 0 \Leftrightarrow X_i \in \mathcal{N}_X(Y)$ where $\mathcal{N}_X$ is the neighbourhood of any point in the dataset X. That is to say $w(Y)$ satisfies the optimization problem:

$$w(Y) = \arg\min_{w \in \mathbb{R}^{N_s}} \|Y - \sum_{i=1}^{N_s} w_i X_i\|_2^2 \text{ s.t. } w_i \neq 0 \Leftrightarrow X_i \in \mathcal{N}_X(Y)$$

As for the extension K_{DM} of the Diffusion Maps kernel, we have:

$$\tilde{K}_{DM}(Y, X_i) = \frac{k(Y, X_i)}{\sqrt{\mathbb{E}_j[k(Y, X_j)]\mathbb{E}_j[k(X_i, X_j)]}} \tag{1.17}$$

with $\mathbb{E}_j[k(Y, X_j)] = \frac{1}{N_s}\sum_p k(Y, X_p)$ and where k is the *diffusion kernel*, *i.e.* the kernel used to provide distances accordingly to the diffusion process. The most widely used kernel and the one that we have been using in this thesis, is the notorious gaussian kernel given in equation 1.18:

$$\forall a, b \in \mathbb{R}^d, k_g(a, b) = \exp\left(\frac{-\|a - b\|_2^2}{2\sigma^2}\right) \tag{1.18}$$

where σ is often called the *bandwidth* parameter.

Now that extensions for each of the kernels have been provided, we can extend the dimension reduction algorithms with the help of Nyström formula:

$$\forall k \in [\![1, m]\!], \tilde{\pi}_k(Y) = \frac{1}{\lambda_k}\sum_{i=1}^{N} \pi_k(X_i)\tilde{K}(Y, X_i) \tag{1.19}$$

Where $X = (X_1, \ldots, X_N)$ is the training set, $\tilde{K}$ is the extended kernel and λ_k the k-th largest eigenvalue in the spectral decomposition of $\tilde{K}$. A short analysis of this equation tells us that that the new embedded point is constructed as a linear combination of the embedded training sample, weighted by the kernel distances from the test point to the training set (see figure 1.15 as an illustration of the out-of-sample extension). This is noteworthy and could be a benefit of the Nyström method, as this will have a strong normalizing effect over anomalous test samples.

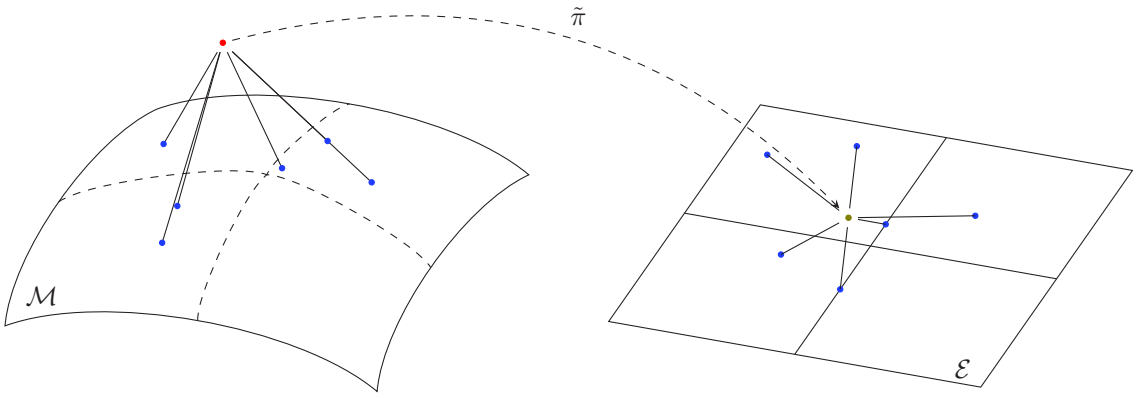

Figure 1.15: An illustration of the out of sample extension step: distances between the new sample (red point) and data points (blue points) are used to extend the kernel and the formula of 1.19 provides the point in the embedding corresponding to the test sample.

The out-of sample problem could also be tackled with other means of regression, as we are trying to predict the embedding of a new test point corresponding to a high dimensional point, a task we already did for the whole training set. We could thus use a supervised regression algorithm (such as linear regression, kernel ridge regression [41] random forest [42], etc.) and train it over our original pairs of high dimensional/low dimensional embedded data. However, none of this methods extrapolate correctly the geometry of our data and the Nyström method has the advantage of statistically converging to a correct estimation of our eigenvectors. It also provides a "normalizing" effect, as it consists in a linear projection over the training embedding. However, it is **not a robust method** of out-of- sample, as it is also strongly affected (*via* the kernel) by anomalies. Indeed, a more abnormal subject will have a more "uniform" kernel, hence a dimension reduction corresponding to the mean of the training embedding.

Figure 1.16 illustrates as an example a comparison between out-of-sample extensions of the Diffusion Maps on a trivial dataset (three quarters of a circle of unit radius, colormap given by the sampled angle). We observe an expected behaviour from the nearest neighbour and linear regressions: the nearest neighbour regression induce a Vornoï tessellation of the space according to the angle of the closest neighbours of each point, while the linear regression only captures one direction of variance for the angle. The Nyström extension, on its side, is much closer to the ground truth with a smoother evolution than the nearest neighbours one, and a smaller "border" between the angle values extrema. However, as can be seen on figure 1.17, if we try to perform an out-of-sample extension with the Nyström method on points farther than a certain threshold (established by the kernel used in the dimension reduction method), the provided extension is only the mean of all training set embeddings, which could

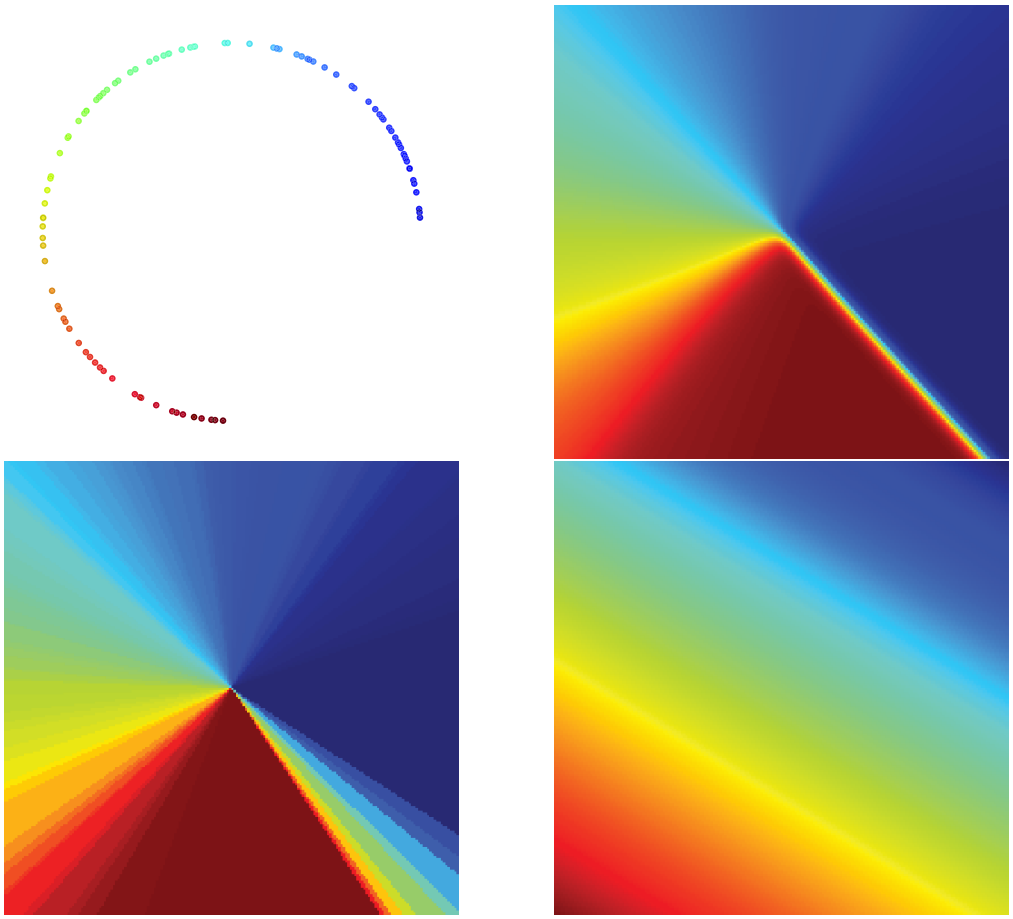

Figure 1.16: A comparison of different out-of sample solutions for the extension of a dimension reduction. Top left presents the original data, coloured by its dimension reduction obtained with isomap, while the other three panels present respectively the extension of: top-right/Nyström, bottom-left/K-neighbours regression, bottom-right/Linear regression. The domain of the represented square is $[-1, 1]^2$

be seen in 1.19: as a point is far from every training samples, its kernel values will all be quite close and thus the linear combination will translate into a mean.

1.3.3 Partial Conclusion

In this section, we have highlighted the need for an **out-of-sample extension** to our dimension reduction techniques, and **have presented a solution** for both the linear one (the inherent extension of PCA), and the non-linear ones with the help of the **Nyström method**, an eigenfunction extension function widely used in kernel methods. We stressed out the **need for our out-of-sample extension to be robust**, and presented a robust method based on the PCA algorithm. Let us now look into the **preimage problem** for non-

Figure 1.17: A farther look (the domain is now $[-5, 5]^2$) of the out-of-sample extension showcased in figure 1.16 for the Nyström method (top-right panel). Points farther than a given threshold have all the same extension corresponding to the mean training set embedding.

linear methods.

1.4 Preimage Problem

Contents for this section

Just as for the out-of-sample problem, non-linear dimension reduction techniques are not equipped with an inner reconstruction method (*i.e.* a method to provide a high-dimensional image - called the preimage- corresponding to a low-dimensional point in the embedding). Finding this preimage is an ill-defined and complex problem as there is no cause for it to exist [17], nor for it to be unique.

As we are not looking for an *exact* preimage but rather an approximation of it (and actually, in the case where an exact analytic way of finding a preimage is available, results with it are worse since an exact preimage does not generally exist [43]), the pre-image problem can be tackled numerically by treating it like a regression problem. However, we will first deal with the special cases of PCA and then of kernel methods, as all the methods we presented are kernel ones.

1.4.1 PCA reconstruction

As we already stated in section 1.2.1 and in similar fashion to the out-of-sample extension from section 1.3.1, PCA is equipped with an inherent reconstruction method, which was given by the trivial formula of equation 1.7 as: $\rho(x_n) = Wx_n + \overline{X}$. This reconstruction is obviously linear, but as opposed to what we will see further on, not directly based on training samples, but instead on the eigenvectors obtained in the principal components decomposition obtains in the training phase.

1.4.2 The Preimage Problem in Kernel Methods

The preimage problem has been extensively addressed in the context of kernel methods, in particular for the kernel PCA [16] preimage problem [44, 17, 45, 43, 46, 47], in the context of image denoising (mainly applied to the notorious MNIST dataset). he peculiarity of [46] being that it connects the problem of finding a high-dimensional pre-image to the one of finding a low-dimensional out-of-sample extension with the Nyström approach. All the preimage methods described in the literature are quite similar, in the sense that they rely on kernel methods properties to provide a solution to the pre-image problem, especially the *definite positiveness* of the kernel being used. The Isomap and LLE kernels, however, are *not* positive definite kernels, and do not apply for a

solution such as the Kernel PCA one. The general form of the solution 1.20 for Kernel PCA is nevertheless a good source of inspiration for finding a solution to the LLE and Isomap problems.

$$\frac{\sum_i \alpha_i K(Y, X_i) X_i}{\sum_i \alpha_i K(Y, X_i)} \tag{1.20}$$

where the α_i denote a probabilistic weighting of the samples.

1.4.3 The Nadaraya-Watson Kernel Regression

The expression given in 1.20 is unmistakeably close to a form of Nadaraya-Watson Kernel Regression [48] (NWKR). NWKR is a non-parametric, unsupervised, statistical method of multivariate regression based on kernel density estimation. Given a kernel K (that can be extended in $\tilde{K}$), the formula for NWKR is the one from equation 1.21. In our context, it can be applied to solving the pre-image problem [49, 50] (both based on [51]) by providing a high-dimensional image corresponding to a low-dimensional embedding point, according to its relationship to the training embedding points with the kernel, and based on the high-dimensional training points.

$$\rho(y) = \frac{\sum_{i=1}^{N} \tilde{K}(X_i, y) X_i}{\sum_{i=1}^{N} \tilde{K}(X_i, y)} \tag{1.21}$$

A commonly used kernel for regression is the gaussian kernel with bandwidth parameter σ. For two sample points X_i, X_j of the original space:

$$K(X_i, X_j) = \exp\left(-\frac{\|\pi(X_i) - \pi(X_j)\|^2}{\sigma}\right) \tag{1.22}$$

And a natural extension arises for a new embedding sample point y:

$$\tilde{K}(X_i, y) = \exp\left(-\frac{\|\pi(X_i) - y\|^2}{\sigma}\right) \tag{1.23}$$

The bandwidth of the gaussian kernel can either be automatically estimated as $\sigma = \text{mean}\{\|\pi(X_i) - \pi(X_j)\|^2\}, \forall i, j \in [\![1, N]\!]\}$, or by cross-correlation over our training set by minimizing the mean squared error of reconstruction.

A great advantage of taking the gaussian kernel is that its exponential decay provides a natural "neighbour selection" for a new sample among the training points in the embedding: numerically, after the kernel evaluation, only points in the embedding sufficiently close to our test points will have non-zero kernel values.

Figure 1.18 illustrates the reconstruction process with kernel regression: first we compute the kernel distances k_i, then we perform a linear combination of all our training set according to the corresponding weights.

Just as for the out-of sample problem, this could be treated as a simple regression problem where we try to predict high dimensional images coming

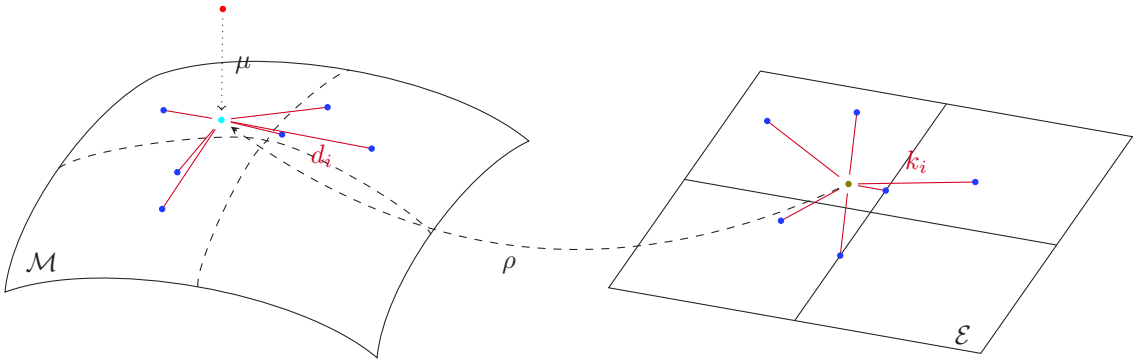

Figure 1.18: Illustration of the preimage computation *via* the Nadaraya-Watson Kernel Regression

from low dimensional embedding, and we can indeed see our solution to this problem as one. However, we will see in the following section that in addition of making a nice link with kernel pre-image methods and being extremely fast to compute, the Nadaraya-Watson kernel regression also fits wonderfully well as a projection method for our anomaly detection.

1.4.4 Diffusion Maps and Original Space Gradient

As previously stated in section 1.2.2, the Diffusion Maps algorithm comes equipped with a diffusion distance in its embedding, which possesses the nice property of being differentiable and, provided that we correctly chose the kernel associated with the algorithm, of being analytically derivable. Here lies the idea exploited in [52] in the context of shape priors to obtain a preimage: try to find a high dimensional image in the original space minimizing the diffusion distance (*i.e.* the euclidean distance in the reduced space) between the test image and the out-of-sample extension of the image we are doing the optimization on. This optimization is non-convex and thus of course highly dependent of the initial guess, but it also provides a much larger search space for reconstruction than the training points convex hull on which we worked in the previous sections.

The major flaw of this method is that the optimization is done in the high dimensional space, and thus is quite computationally intensive and difficult to do. Equation 1.24 presents the optimization that needs to be solved to obtain a preimage.

$$\mu(Y) = \arg\min_{Z} \|\tilde{\pi}(Z) - \tilde{\pi}(Y)\| \tag{1.24}$$

with some constraint to avoid trivial solutions, such as:

$$Z \in \mathcal{C}_X(Y)$$

where $\mathcal{C}_X(Y)$ is the convex hull formed by the closest neighbours of Y in the

training set X. More details on the reconstruction problem for the DM can be found in the dedicated section 2.3.2.

1.4.5 Partial Conclusion

In this section, we presented several approaches to perform a reconstruction from the low-dimensional embedding to the ambient space, also called the **pre-image problem**. As it is an **ill-posed** problem, we established that we would look for an approximate pre-image using numerical methods.

This problem has been dealt with for the different situations corresponding to each class of algorithms:

- In the case of PCA, a **linear reconstruction** based on the principal components is inherently part of the algorithm.
- For **kernel methods** with a positive definite kernel, various methods of obtaining a preimage can be obtained, involving **optimization schemes**.
- For non-positive definite kernels such as **Isomap or LLE**, a reconstruction inspired by the positive definite case can be computed, using the **Nadaray-Watson kernel regression**.
- Finally for the **Diffusion Maps**, as it provides an embedding distance differentiable in the ambient space, an **optimization procedure** allows for a pre-image computation.

1.5 Deep Learning Methods

Over the course of recent years, deep learning techniques have taken an increasing place in computer vision methods (and publications). With a turning point being the introduction of the backpropagation algorithm [53, 54] and help from the rise of computing power nearly following Moore's law in Graphics Processing Units, neural network based algorithms (that can be traced all the way back to the 1950s [55, 56]) have skyrocketed to undoubtedly become the most popular machine learning techniques nowadays. With extremely notorious applications in seemingly simple tasks such as character recognition [57] or image recognition [58], but also in fields closer to our paradigm, such as image synthesis with the successive apparition of Deep Belief Networks [59] (DBN), Restricted Boltzmann machine [60] (RBM) and Generative Adversarial Networks[61, 62] (GANs), the latter being now one of the gold standard for audio or image synthesis. In relation to our domain, some deep learning techniques have had a tremendous success [63], and are now extremely popular.

An observant reader should have noticed at this point the closeness between our paradigm and a specific class of neural networks being auto-encoders. These deep learning algorithms provide multiple representations of our data with decreasing dimensionality, with non-linear relationships from one representation to the next, leading to a final low-dimensional representation of our data which is reconstructed step by step with representations of increasing dimensionality into a data point of dimensionality equal to the original point. Thus, synthetically an auto-encoder consists in a non-linear dimension reduction step, and a non-linear reconstruction one, which is indeed close to the projection paradigm we are using. In relation to the paradigm, auto-encoders have the amazing properties of having not only the out-of-sample extension but also the reconstruction built-in with the dimension reduction into only one method. Starting from a simple dense, feed-forward neural network with a single intermediate layer and "linear activation functions" (which can be shown to be equivalent to the modelization of PCA [64, 65]), autoencoders have evolved into deep networks with complex layers and modelizations, for instance sparse [66], variational [67] or convolutional [68] autoencoders.

Nevertheless, to the best of our knowing, neither auto-encoders nor GANs are yet used to perform anomaly detection in the same sense that we are doing in a 3D setting. While we would not provide a reason for real researchers, we should try to explain why we did not use this kind of techniques ourselves. First, deep learning algorithms and especially the recent kinds of deep convolutional autoencoders/GANs are incredible algorithms, capable of learning extremely complex models over non-linearly distributed dataset in a much more precise manner than many previous machine learning algorithms. In order to learn a meaningful model however, they do need an according number of training samples. The most impressive deep learning results over *classical* machine learning have been obtained with (extremely) large number of samples

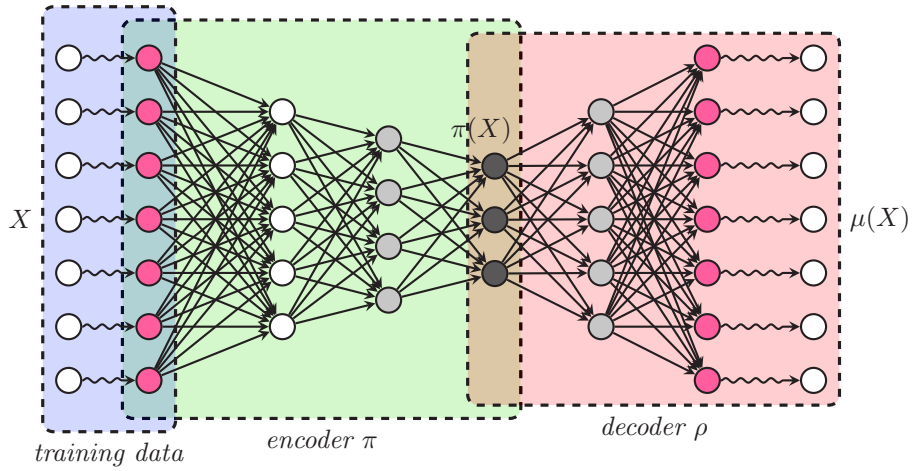

Figure 1.19: A graphical model of a dense, auto-associated autoencoder. Starting from the original data, an encoding phase compresses the data using linear combination of neurons from the previous layer and non-linear activation functions, until the "embedding layer" (in dark grey), from which a decoding phase decompresses the data in similar fashion until reaching the original data dimension.

(sometimes in the order of millions), or with dataset that can be artificially "augmented". Data augmentation is a technique consisting on expending the training set based on the samples we already have. They usually rely on geometric transformations, such as rotations or jittering (small translations in various directions), in order to make the model robust to these transformations. One other way of augmenting the data in a sense, is to consider only parts of it and create a problem with a smaller dimensionality with a drastically increased number of samples (for instance by working with patches) . In this thesis, the number of training samples originally available was around 30, and has slowly increased along the way to reach about 1500 samples which is still not near what can be considered a large dataset, especially considering the dimensionality of the problem (nearly 10 millions of voxels in the brain a full-size MRI). Data augmentation using geometric transformations is not a solution for us as we are dealing with registered data. Neither are local approaches such as patch, as we want our model to capture global correlations in our data (such as left-right brain symmetry). Therefore we are stuck in a few samples/high-dimensionality setup, for which deep learning techniques are not really famous.

A second problem is computational. As we previously mentioned, we are dealing with 3D data with a huge count of voxels. Deep learning algorithms (especially convolutional ones that are widely used in image processing) tend to be greedy in the number of parameters used in the model and thereby in memory usage for the GPU used to train it. This is a problem considering most of the available GPUs are limited to 8GB of VRAM. To alleviate this

issue, some methods using deep learning over 3D datasets either use patches approaches, or 2.5D ones where only a small number of image slices are considered at once. As we already asserted, we unfortunately cannot use this kind of techniques with our approach.

For these reasons, we decided to stick with the "conventional" machine learning methods described in sections 1.2 to 1.4, and create the anomaly detection algorithms that we will present in the following section with them.

Chapter 2

Contributions

This Chapter contains:

This chapter is dedicated to presenting anomaly detection methods based on the paradigm and various methods we presented in chapter 1. After a brief section on the usage of statistical testing for anomaly detection, we will move along with our second section 2.2, that will be consecrated to state-of-the-art methods of anomaly detection used in our voxel-wise medical images context of detecting pathology afflicted areas. The following section 2.3 will introduce anomaly detection methods using the tools provided in the previous section, where we provided few contributions but that were either not used as anomaly detection methods, or not in this context. The last section 2.4 will be focused on methods using a new framework of manifold projection, using kernel methods.

2.1 Z-Score for Anomaly Detection

In order to provide anomaly detection methods that can be backed up with *probabilities* or *scores* of detection, we will add a final step to our algorithms of projection (the first one being the computation of the residuals 2.1). This final step (see figure 2.2) consists in computing a Z-score at each voxel of our images based on the residuals between our original test image and the high-dimensional projected data (whatever the projection method).

First, for each voxel, we will learn the distribution of the residuals R_i under the null hypothesis $\mathcal{H}_0$ that no anomaly is present in this voxel. This distribution will be modelled as a gaussian distribution with 0 mean (residuals are centred under $\mathcal{H}_0$). Therefore the variance at each voxel can be computed as: $\sigma^2 = \frac{1}{N}\sum_i R_i^2$ (here we omit notations for voxels to make it less cluttered, and present it as in a one dimensional setting). To avoid any bias, we do not estimate the variance of the distribution of residuals on the training set (that we only use to learn the model of projection), but rather on what we call a validation set (denoted V): normal samples that have been put aside from the training set in this purpose and are therefore not used to learn the projection.

Finally, the Z-score associated to a test subject can be computed as:

$$Z_{test} = \frac{R_{test}}{\sigma_V} \tag{2.1}$$

Where σ_V is the standard deviation over set V. This Z-score computation will be the ending step of all our methods of projection, thus completing the transition of our projection paradigm into a voxel-wise anomaly detection one. All methods of projection will therefore also be considered anomaly detection ones, as long as they provide a projection for any test sample. A most "normal" voxel would correspond to a null Z-score, while a more suspect one would have a bigger Z-score in magnitude (either strongly negative, or strongly positive).

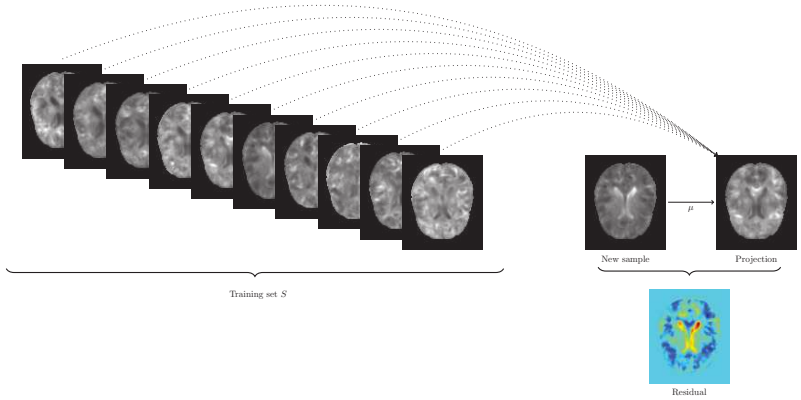

Figure 2.1: Residual computation for a test sample, based on its projection μ.

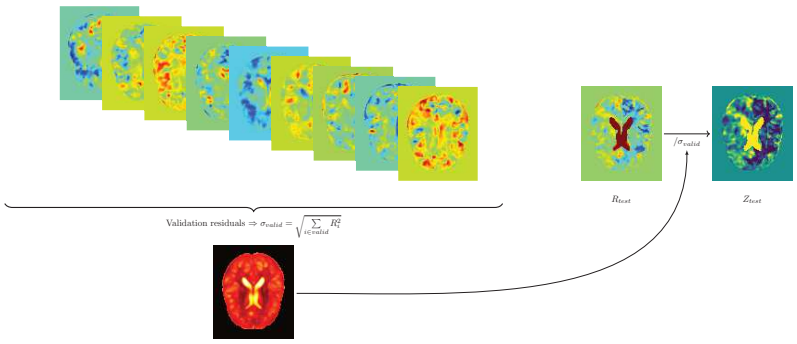

Figure 2.2: Z-score computation of a test residual against "validation" residuals under the null hypothesis that residuals are centred.

2.2 Linear Methods of Anomaly Detection

Contents for this section

In this section, we will consider methods of voxel-wise anomaly detection/high-dimensional projection that are both linear and state-of-the-art in voxel-wise anomaly detection for medical images. As such, they are prime comparison points to the algorithms we developed on our own and that we will present later on.

2.2.1 Global Linear Model for Anomaly Detection

The general linear model (GLM) is a statistical framework including methods of hypothesis testing such as the analyse of variance (ANOVA [69]), the analyse of covariance (ANCOVA [70]) and their multivariate counterparts (MANOVA [71], MANCOVA). The GLM has been first used in medical imaging with its most famous Matlab implementation, *Statistical Parametric Mapping* [72, 73] and both now make reference in medical image analysis fields such as voxel-based morphometry [74, 75, 76, 77], which looks for volume regressions or dilatations into the brain and is concurrently one of our focuses. It is *widely* used in the medical community as a framework for statistical group comparison (pathological vs healthy essentially) in order to statistically delimit abnormal areas and therefore spot which organs are most susceptible to be affected by the pathology. This in turn, gives doctors insights into the pathology behaviour [76, 77].

The GLM can be written as:

$$X = D\beta + \varepsilon \tag{2.2}$$

where X is our complete dataset of samples (possibly with several groups, with N total samples), D a design matrix of *explanatory variables*, β a vector of r unknown parameters or *regressors* that are used as coefficients for the explanatory variables, and ε the noise variable, with usually $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$ under the *homostedasticity* hypothesis.

Remark: The GLM model is a univariate one, each statistical test is done independently on each voxel. Therefore for readability purposes, we will drop any voxel- wise notation. Please bare in mind however that all the presented equations are derived for "one-dimensional" samples (each voxel).

Under the homostedasticity hypothesis, the optimal least-square estimation for β gives us:

$$\hat{\beta} = (D^T D)^{-1} D^T X \tag{2.3}$$

and is an unbiased estimator of β, with variance-covariance matrix:

$$\Sigma_\beta = \sigma^2 (D^T D)^{-1} \tag{2.4}$$

provided that $D^T D$ is invertible, and thus that D is a full rank matrix.

As σ is generally unknown, it is also estimated: $\hat{\sigma}^2 = \frac{\varepsilon^T \varepsilon}{N-r}$. Whereof we get that the covariance matrix of $\hat{\beta}$ can be estimated as:

$$\hat{\Sigma}_\beta = \frac{\varepsilon^T \varepsilon (D^T D)^{-1}}{N-r} \tag{2.5}$$

A common use case of the GLM in statistical group comparison is performing statistical tests between two groups, which is remarkably easy to do in this framework. For instance, assuming we would like to perform the most classical Student's t-test of equal mean between two populations X_1 and X_2 (of respective sizes N_1 and N_2), we could set up a GLM expression of this t-test by using the following design matrix:

$$D = \begin{matrix} \\ N_1 \\ \\ \\ \\ N_2 \\ \\ \end{matrix} \left\{ \left(\begin{array}{c|c} 1 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ \hline 0 & 1 \\ \vdots & \vdots \\ 0 & 1 \end{array} \right) \right.$$

With this design matrix we trivially derive the following steps: $D^T D = \begin{pmatrix} N_1 & 0 \\ 0 & N_2 \end{pmatrix}$, from where it simply follows that

$$(D^T D)^{-1} D^T = \left(\begin{array}{ccc|ccc} 1/N_1 & \dots & 1/N_1 & 0 & \dots & 0 \\ \hline 0 & \dots & 0 & 1/N_2 & \dots & 1/N_2 \end{array} \right) \begin{matrix} \\ \underbrace{\qquad\qquad\qquad\qquad\qquad\qquad} \\ N_1 \qquad\qquad\qquad N_2 \end{matrix}$$

and finally the estimation of β:

$$\hat{\beta} = (\overline{X_1}, \overline{X_2})^T \tag{2.6}$$

A statistical test is then derived using a *contrast* vector c: under the *null hypothesis* $\mathcal{H}_0$, a contrast vector is used to confront a linear combination of our regressors to 0, that is to say under $\mathcal{H}_0$, $c^T \beta = 0$. As c is fixed, the estimator

for $c^T\beta$ is naturally $c^T\hat{\beta}$, while the estimated variance of $c^T\beta$ is given by $c^T\hat{\Sigma}_\beta c$. Ultimately, under $\mathcal{H}_0$,

$$\frac{c^T\hat{\beta}}{\sqrt{c^T\hat{\Sigma}_\beta c}} \sim t_{N-r} \tag{2.7}$$

where t_k is the Student's law with k degrees of freedom, which rapidly converges to a centred, reduced, normal law as k tends to fairly large values (100 samples is large enough to consider t_{100} being the normal law), which is usually the case when dealing with group analysis. Using a $\begin{pmatrix}1\\-1\end{pmatrix}$ contrast with the previously defined contrast matrix D, we derive a statistical test designed to check for equal means between the two populations under $\mathcal{H}_0$. Indeed, the estimator of the "contrasted regressors" is $c^T\hat{\beta} = \hat{\beta}_1 - \hat{\beta}_2$ and once the associated variance has been computed, starting from equation 2.5 and coupling it with equation 2.7, we get that under $\mathcal{H}_0$:

$$\frac{\hat{\beta}_1 - \hat{\beta}_2}{\sqrt{\varepsilon^T\varepsilon(\frac{1}{N_1}+\frac{1}{N_2})/(N_1+N_2-2)}} \sim t_{N_1+N_2-2}$$

with the equation of the GLM 2.2.1, $r = 2$ and the choice of equation 2.2.1 for D, we have for ε:

$$\varepsilon = \begin{pmatrix} X_1^T - \overline{X}_1 \\ \hline X_2^T - \overline{X}_2 \end{pmatrix}$$

and therefore:

$$\varepsilon^T\varepsilon = \begin{pmatrix} X_1^T - \overline{X}_1 \\ \hline X_2^T - \overline{X}_2 \end{pmatrix}^T \begin{pmatrix} X_1 - \overline{X}_1 & X_2 - \overline{X}_2 \end{pmatrix} \begin{pmatrix} (X_1 - \overline{X}_1)(X_1 - \overline{X}_1)^T \\ +(X_2 - \overline{X}_2)(X_2 - \overline{X}_2)^T \end{pmatrix}$$

Let us denote respectively $s^2_{X_1}$ and $s^2_{X_2}$ the unbiased estimators of variance for X_1 and X_2. $s^2_{X_1}$ and $s^2_{X_2}$ are defined as such:

$$s^2_{X_1} = \frac{1}{N_1 - 1}\sum_{i=1}^{N_1-1}((X_1)_i - \overline{X}_1)^2$$

$$s^2_{X_2} = \frac{1}{N_2 - 1} \sum_{i=1}^{N_2-1} ((X_2)_i - \overline{X}_2)^2$$

which is very close to the expressions found in $\varepsilon^T\varepsilon$ (recall here that we are doing a voxel-wise analysis). Rewriting the expression of $\varepsilon^T\varepsilon$ with $s^2_{X_1}$ and $s^2_{X_2}$, we get:

$$\varepsilon^T\varepsilon = (N_1 - 1)s^2_{X_1} + (N_2 - 1)s^2_{X_2}$$

Therefore, if we define s^2_P as the *pooled variance* of X_1 and X_2, we have:

$$\varepsilon^T\varepsilon = (N_1 + N_2 - 2)s^2_P$$

which in turns, yields a simplified expression for equation 2.2.1:

$$\frac{\hat{\beta}_1 - \hat{\beta}_2}{s_P\sqrt{\frac{1}{N_1} + \frac{1}{N_2}}} \sim t_{N_1+N_2-2} \tag{2.8}$$

This is precisely the expression for a two-sampled t-test with same variance aimed at testing for equal means between X_1 and X_2 under homostedasticity hypothesis, with the general approximation being that this Student distribution is actually a gaussian one. This demonstrates the ability of the GLM to easily perform (one of) the most classical statistical analysis of our dataset.

The GLM is also used in a somehow degenerate case to provide an anomaly detection in the single subject vs group setting (ours). In this setup, an individual sample Y is confronted to a group by performing the same statistical test (*i.e.* Student's t-test mean comparison between two groups). By considering the second group to be constituted with only the individual to be tested (and therefore that $N_2 = 1$ and $S_{X_2} = 0$), we derive a functional -albeit sketchy- test to perform anomaly detection on a single subject compared to a normal group (the obtained p-values or t-statistics can be thresholded to obtain a binary voxel-wise detection): starting from equation 2.8, we get:

$$\frac{\hat{\beta}_1 - Y}{s_{X_1}\sqrt{\frac{1}{N_1} + 1}} \sim t_{N_1-1} \tag{2.9}$$

with the approximations resulting from large enough values of N_1, we obtain the associated expression:

$$\frac{\overline{X}_1 - Y}{s_{X_1}} \sim \mathcal{N}(0, 1) \tag{2.10}$$

which coincidentally is the same expression as the one we used for our own methods from equation 2.1, where the projection of a any test sample Y is constant and equal to the mean of the training sample:

$$\mu_{GLM}(Y) = \overline{X}_1 \tag{2.11}$$

with the slight difference that in our case the variance $s^2_{X_1}$ is learned on an independent dataset of validation samples. We stated that the difference between the *true* modelization of the GLM, that we just described, and the one using our paradigm with the projection 2.11 was negligible, and that we would rather have all the tested methods belong to the same paradigm. We will thus refer further to the method obtained by projecting each sample onto the mean of the training sample as **GLM**.

2.2.2 PCA-Based Methods for Anomaly Detection

The PCA algorithm presented in section 1.2.1 also directly translates into an anomaly detection method. As it provides a high-dimensional projection/normalization of our subject, we only need to perform a Z-score computation over our test residuals, as mentioned in section 1.1. For future reference, we will denote as **PCA** the anomaly detection algorithm derived from the Z-score computation carried out in equation 2.1 over the projection obtained with the PCA algorithm. On a similar point, we will denote **RPCA** the one we get with the robust counterpart of PCA introduced in section 1.3.1.

2.2.3 Conclusion

In this section, we reviewed the main linear, state-of-the-art methods for voxel-wise anomaly detection in images (or equivalently the projection ones, as stated by our paradigm).

- The most frequently used one in a medical context being the **GLM** and its implementation **SPM**. While SPM is more frequently used for group comparison rather than subject-vs-group analysis, it provides a powerful but flawed tool for our purposes and a challenging contestant for our methods.
- We also presented the *extremely classic* method of projection (and therefore anomaly detection) derived from the **PCA** algorithm, and its robust version the **RPCA**.

We will now shift our attention to more original methods, that are not completely innovative in the sense that a method close to the ones described in this work has already been presented in the literature, but never in a context of anomaly detection.

2.3 Dimension Reduction-Based Non-Linear Methods

Contents for this section

2.3.1 Nadara-Watson Kernel Regression for Reconstruction

As we previously addressed in section 1.4.3, NWKR can be used to solve the preimage problem for any dimension reduction techniques deprived of an inherent preimage method, thus completing the projection algorithm needed to achieve a full high-dimensional projection. As we have seen in the previous section with the example of PCA, this projection is all that is needed to perform anomaly detection, as statistical testing is carried out on the ensuing residuals (*i.e.* with no step in between). In the case of Isomap, in [51] and [49] authors present a way to provide a preimage for training samples using NWKR and use it as a generative model [49] by sampling in the embedding space. But their modelization is different from ours in the sense that they only address the problem of generating new samples: they provide an embedding for the training set and sample new elements by reconstructing sampled points from the embedding with NWKR. They do not deal with any test sample, and therefore do not compute any out-of-sample extension of a new sample outside the training set, let alone its high-dimensional projection.

Hence, to the best of our knowledge, the method of projecting a test sample with the help of the Nyström out-of-sample extension and the NWKR, is original and our own. We have yet however to demonstrate its utility in the field of voxel-wise anomaly detection. For the rest of this thesis, we will denote the anomaly detection based on the Isomap dimension reduction with Nyström out-of-sample extension and NWKR preimage as the **ISO** method. We will *not* use NWKR to provide a preimage for LLE or the diffusion maps, as better options are available in both cases.

We will perform our NWKR reconstruction using a gaussian kernel 1.18, for which we need to provide the bandwidth parameter. This is done by cross-validation: we optimize the bandwidth parameter on the reconstruction of normal samples from the training set by classically splitting it in folds, one of the fold being a testing set while the others are used for training, and cycling these roles around all the folds. The resulting bandwidth parameter is the one that minimizes the mean MSE across all the permutations.

2.3.2 Diffusion Maps and Original Space Gradient

As we previously introduced in section 1.2.2, the Diffusion Maps algorithm can be used as a direct method of projection (and therefore an anomaly detection method) by using an optimization scheme. This optimization scheme focuses on minimizing (with constraints) the diffusion distance (*i.e.* the embedding space distance) between our test sample and its projection. Recall the objective function that is used with a convex hull constraint to perform the pre-image computation (the out-of-sample extension being the one of the Nyström extension 1.3.2):

$$\mu(Y) = \arg\min_{Z} \|\tilde{\pi}(Z) - \tilde{\pi}(Y)\| \quad \text{s.t. } Z \in \mathcal{C}_X(Y) \tag{2.12}$$

The idea of performing the projection of a test sample over a previously learned manifold is not ours, but rather the one of [52, 78, 79, 80], where the authors describe a novel method based on the Diffusion Maps algorithm that aims at performing what is called the (weighted) *karcher mean* of points in a manifold, that is to say the *geodesic* mean of such points: the point in the manifold evenly close to all the points over which we are performing the mean. This karcher mean is then used, either to interpolate a shape (not originally present in the training set, and presenting alterations such as occlusions) using its diffusion distance neighbours, or to denoise a manifold of digits images. This method was applied to occluded 3D medical shapes, and was initially aimed only to shape datasets, therefore using tools specific to this domain of research. We did not come up with any of the ideas presented in these articles, but neither did we apply them directly to shape manifolds: we had to adapt the method to our own manifold of medical images and thus drop any use of shape analysis tools. In [80], a variational formulation of the problem is built around an energy that can classically be split into a prior term and an active contour term. After establishing the solution of the variational approach as the solution of a constrained optimization, they define an iterative method to solve this optimization problem in which the tangent space to the space of acceptable solution is computed. The gradient of the cost function at each iteration is projected over this tangent space in order to ensure that, starting from a point in the space of acceptable solutions, the intermediate points considered by the algorithm until it reaches convergence are also acceptable solutions.

The following equation 2.13 presents the optimization problem we derived to perform our own manifold projection, that is directly inspired by the work we just introduced (albeit at the expense of the tangent space computation):

$$\begin{aligned} \mu(Y) &= \sum_{i \in \mathcal{N}_x(y)} \hat{\theta}_i X_i \\ \text{s.t. } \hat{\theta} &= \arg\min_{\theta} \|\tilde{\pi}(\sum_{i \in \mathcal{N}_x(Y)} \theta_i X_i) - \tilde{\pi}(Y)\|_2 \\ \text{and } \theta &\succcurlyeq 0, \ \sum_i \theta_i = 1 \end{aligned} \tag{2.13}$$

where $\mathcal{N}_x(Y)$ is the indices of the closest neighbours of Y in the embedding. Here we take advantage of the advanced modelization of the diffusion maps by looking for a projection that is *still* a linear combination of training samples, but relying on the diffusion distance (and not the one in the original space) between our test sample embedding and the embedded point of the variable solution. We thus ensure our solution is both close to the original point in ambient and embedding spaces.

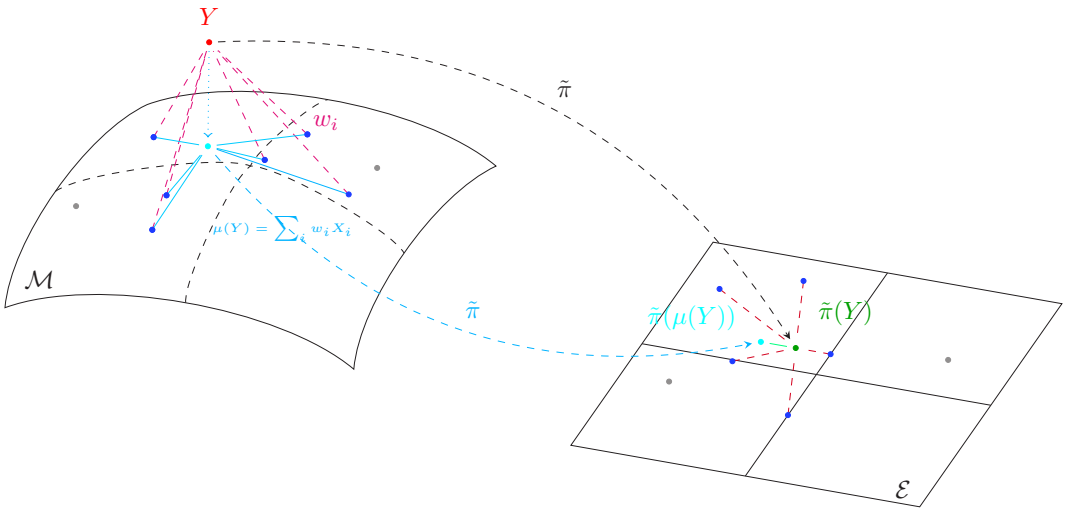

Figure 2.3: An illustration of the optimization solution for the manifold projection technique using the diffusion maps modelization. The algorithm looks for a point in the original space $\mathcal{M}$ (in cyan) that has a corresponding point in the embedding $\mathcal{E}$ as close to the one of the test sample as possible, while being expressed as a barycentric combination of the test sample Y neighbours (the neighbours being chosen among the learning samples X_i, but according to the embedding distance).

As before, we will refer to the anomaly detection method resulting from this projection method as its inherent dimension reduction method: **DM** (Diffusion Maps).

2.3.3 Projection by out-of-sample optimization

In the previous methods, our reconstruction was determined most of the times by the out-of sample extension being used (*i.e.* most generally the Nyström extension 1.3.2) to extend our mapping π from the training set to our test sample. This obviously highly relies on the result of the Nyström method, while being extremely constrained: the Nyström extension is expressed as a weighted linear combination of the low dimensional embeddings of our training samples. This could potentially be performance-impairing, as it prevents us from exploiting the full low dimensional embedding as search space for a projection of our test sample (there is *a priori* no cause for any given test point low dimensional embedding to be expressed as such). We will not try here to construe whether the Nyström extension is good or bad for our own projection problem, but rather to introduce a new optimization scheme that aims at tackling both the out-of-sample and preimage problems together, while obtaining a hopefully better reconstructed image. The reconstructed image will still be obtained *via* the NWKR, in the same way that it would be done with the algorithm described in section 2.3.1, but the out-of-sample extension from which the image is reconstructed will not be the one of the Nyström method.

The idea behind this optimization scheme is, starting from any dimension reduction method, to find the *best* low dimensional embedding point corresponding to a high dimensional one, where *best* refers to the one minimizing the mean squared reconstruction error by the Nadaraya-Watson kernel regression. It can be written as follows:

$$\tilde{\pi}_{optim}(Y) = \arg\min_{y} \|\rho_{NW}(y) - Y\|_2$$

where $\rho_{NW}(y)$ refers to the Nadaraya-Watson reconstruction of y, based on the kernel distances from the training set embedding, and over the train set samples. This in turn yields for the projection μ over the manifold:

$$\mu(Y) = \rho_{NW}(\tilde{\pi}_{optim}(Y)) \tag{2.14}$$

This optimisation scheme is non-convex, and is thus susceptible to fall in local minima. But the objective function we defined is easily differentiable, and thus solutions can be obtained fast enough by any minimization algorithm, even with multiple starts to avoid local minima. It is noteworthy that this is still a heavily constrained model as it enforces that our weights w must be consistent with the kernel relationship of our out-of-sample extension with other embedding data points, but it is still less constrained than requiring the dimension reduction of Y to be the Nyström one. We denote **Isoptim** the projection obtained with this optimization scheme and using Isomap as a dimension reduction algorithm.

2.3.4 Partial Conclusion

In this section, we presented methods of manifold projection that previously existed, or were adapted from pre-existing methods from other fields, but that had not been used before in our context of learning a manifold of normality and project a new sample over it.

- The first method that we presented is the one built over the association of the **Isomap** dimension reduction, the **Nyström out-of-sample extension**, and the **Nadaraya-Watson kernel regression** for pre-image reconstruction. Associating these three elements allows for a complete method of projection of *any new sample* over the training set inherent manifold. We called this method of anomaly detection **ISO**.

- A second one, refered to as **DM** is obtained by using the non-linear modelization of the **Diffusion Maps** algorithm 1.2.2. We look here for a projection that can be written as a positive, linear combination of training samples. For the reconstructed point, the **optimization is done in the embedding**.

- The last one is also based on an **optimization in the embedding space** that gets rid of the Nyström extension to attempt to find the best low-dimension point that has the **closest NWKR reconstruction** to our test point. Associated with the Isomap dimension reduction algorithm, we get the **Isoptim** method.

The last section of this chapter is dedicated to presenting techniques for anomaly detection that are not using the embedding provided by the presented original dimension reduction algorithms, but are still using the idea developed in these algorithms to perform a direct manifold projection.

2.4 Projection-Based Methods for Anomaly Detection

Contents for this section

In this section we will introduce original methods for *direct* manifold projection that have been developed during this work. These methods provide a new way of performing either the projection task (finding μ), while not making use of the embedding (the π function) provided by dimension reduction algorithms (nor its out-of-sample extension $\tilde{\pi}$).

2.4.1 Locally Linear Projection

The simplest approach for reconstruction, whether we talk about preimage (coming from a low dimensional embedding such as Isomap) or projection (from the original high dimensional space), is to express the reconstructed sample as a linear combination of the training samples, *i.e.* $\rho(Y) = \sum_{i=0}^{N} w_i(Y)X_i$. Without constraints however, this reconstruction is prone to overfitting. Furthermore, the less we constraint w, the less we corroborate with our manifold hypothesis. In this sections we will present a novel method for high-dimensional projection of any test sample point over a manifold of control samples with what we call *locally linear projection*, or **LLP** (after this section, we will also refer to it as its eponymic dimension reduction method from section 1.2.2, **LLE**).

LLE is the method which modelization is the closest to the manifold definition: indeed, the manifold hypothesis states that locally, the topological space in which our samples lie is identified as a euclidean space (*i.e* a linear one). With locally linear projection however, we will perform no dimension reduction and obtain no low-dimensional subspace. The idea of this method is to come by our high-dimensional projection by using the core concept of LLE: local weights, minimizing L_2 error. While LLE attempted to create a low-dimensional subspace in which local relationships between original space samples were preserved, we merely find a corresponding high-dimensional image to our test image that is as respectful to the local relationships it has with the training set (*i.e.* the weights) as possible, while looking for it as a linear combination of training set samples. Simply put, locally linear projection amounts to find the same weights as LLE and use them to perform a linear

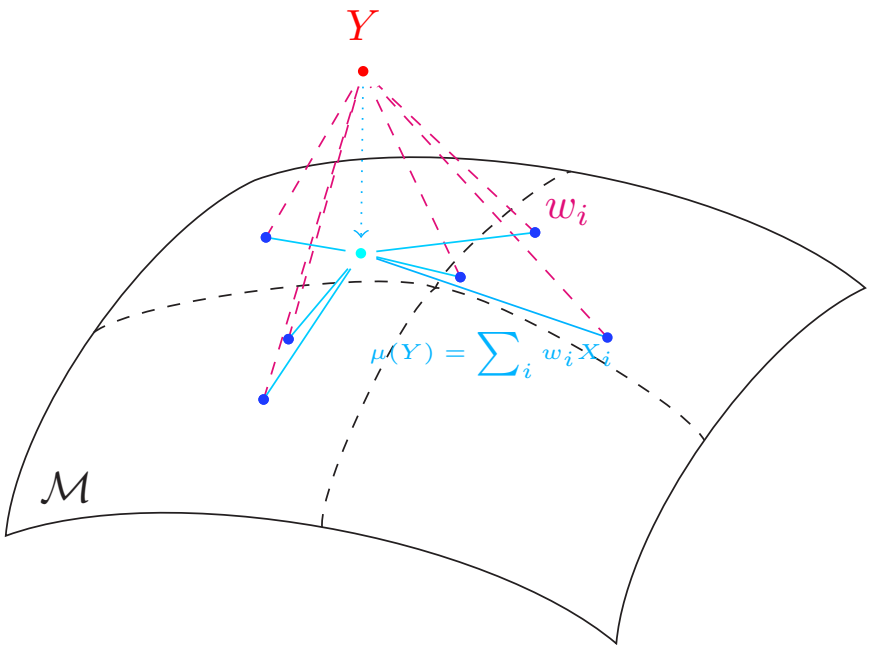

Figure 2.4: A graphical representation of the locally linear projection: A test point (in red) is projected onto the manifold $\mathcal{M}$ by using weights w_i learned over its set of neighbours in the training set (blue points). The resulting projection (in cyan) is obtained *via* a linear combination of the red points using the weights w_i.

combination of training samples:

$$\begin{aligned} \mu_{LLP}(Y) =& X \arg\min_w \|Xw - Y\|_2, \\ & \text{s.t. } \sum_i w_i = 1, w_i \neq 0 \Leftrightarrow i \in \mathcal{N}_X(Y) \end{aligned} \tag{2.15}$$

Just as for the LLE algorithm (it is the same algorithm up until now), the optimization for the weights has a closed-form solution $w_i = \frac{\sum_j C_{ij}^{-1}}{\sum_{k,l} C_{kl}^{-1}}$, where C is the local covariance matrix: $C_{ij} = (Y - X_i)^T (Y - X_j)$. As mentioned, once we have found the weights w, the resulting projection is just the linear combination $\mu(Y) = \sum_i w_i X_i = Xw$ (see figure 2.4 for a graphical representation of the method). Ideally, this linear combination of Y local neighbours belongs to the manifold $\mathcal{M}$ and is close to the projection of Y onto $\mathcal{M}$ that we are looking for. As is occasionally done in implementations of LLE, we added to the multiple constraints of w a L_2 regularization constraint, to alleviate any nearest-neighbour-like overfitting issues, where the weights have the form of a canonical basis vector with a single non-zero, unit coefficient, corresponding to the index of the nearest-neighbour of our test sample (*i.e.* we enforce a much smoother repartition of the coefficients across all neighbours). The ensuing

equations for w is the following one:

$$\begin{aligned}\mu_{LLP}(Y) =& X \arg\min_{w} \|Xw - Y\|_2 + \lambda\|w\|_2 \\ & \text{s.t. } \sum_i w_i = 1, w_i \neq 0 \Leftrightarrow i \in \mathcal{N}_X(Y)\end{aligned} \tag{2.16}$$

This constrained optimization problem seems familiar if we recall the expression of the DM problem from 2.13: indeed, in DM we looked for our projection as being the closest to the test point embedding while being expressed as a weighted sum of training samples. But while DM tried to minimize the diffusion distance (*i.e.* the distance in the embedding space), LLP is focusing on the L_2 distance in the *original* space.

The closest algorithm to LLP we were able to find in the literature is the locally linear (weighted) regression [81] which is mainly used as a non-parametric multivariate smoothing of scattered datasets, and thus completely foreign to any anomaly detection by projection scheme. While not fully original (as it is completely based on the ideas and methods behind the LLE algorithm), LLP is original in the sense that it provides not an embedding but a projection, that will later be used in an anomaly detection scheme, both of which have not (to the best of our knowledge again) been previously developed.

Remark: The already heavy constraints imposed on w can be be made even heavier by adding a convex constraint over w: $w \succcurlyeq 0$ as is often found in the literature (in this case, the w are the test point barycentric coordinates over the set of its closest neighbours). This is the heaviest constraint for w: a local, convex one. It requires for w to be element-wise non-negative, to have a L_1 norm of 1, and to be expressed only over the set of closest L_2 neighbours of our test point in the training set. However, the results we obtained with this additional constraint were not as good as without. We thus stuck to the set of constraints presented in equation 2.16.

2.4.2 Kernel Manifold Projection

The main idea of what we call *kernel manifold projection* (**KMP**), is to define a kernel method that is able to properly model the underlying manifold of our training data, and to provide a projection method onto this manifold. We performed the first task using the fact that many (if not all) the dimension reduction techniques we presented in section 1.2 are actually kernel methods. Indeed Isomap, LLE, and the Diffusion Maps are all relying on the use of a kernel to capture the relationships between sample points (and with the help of the Nyström extension, with *any* point) in their own, non-linear fashion: Isomap is using what is called a *geodesic kernel*, with its approximation of geodesic distances. LLE is using what we could call a *neighbourhood kernel* with its local L_2 reconstruction weigths. Finally, the Diffusion Maps are using a *diffusion kernel*, built upon the symmetrized laplacian graph of kernelized

L_2 distances. The three of them are computing eigenvectors of the "Gram matrices" sampled from their kernel over training and test samples to perform a dimension reduction by providing a low-dimensional embedding that respects the modelization they performed in the high-dimensional space (with the help of some form of MDS). Our idea here is to skip this step of dimension reduction, as we are not *directly* focused on obtaining a low-dimensional embedding, but rather performing a correct non-linear modelization of the training samples manifold, which is done by these algorithms when they compute the Gram matrices before they determine the embedding. These Gram matrices are what we rely on to perform our manifold projection. We inject them straight into the Nadaraya-Watson kernel regression to obtain a projection that is a linear (even convex) projection of the original samples (as most of our projection methods), with weights that are depending on the kernel that is being used to model our data. In practice, if we denote K the Gram matrix of the kernel being used over the training samples, and $\tilde{K}$ its extension to test samples obtained with the Nyström extension, our projection of a test sample can be written as:

$$\mu(Y) = \frac{\sum_i \tilde{K}(X_i, Y) X_i}{\sum_i \tilde{K}(X_i, Y)} \tag{2.17}$$

Remark: This idea is (obviously) inspired by what we are already doing with the Isomap method, but is motivated by the fact that when we perform a reconstruction from the embedding with NWKR, we merely compute distances in the embedding space that are supposed to be good approximations of the geodesic distances between our points. We never use the dimension reduction points for anything else than distance computation. Nevertheless, those embedding points are actually *themselves* computed to respect what was our initial approximation of the geodesic distances with the kernel obtained with Dijkstra's algorithm, or the diffusion kernel. Therefore, the idea here is to avoid performing an approximation of the approximation, and to avoid having to perform the Nyström out-of-sample extension reduction step, which performs *yet again* an approximation for the testing samples. Another important point to be made is that LLP is actually a KMP method for the "locally linear kernel", where said kernel is computed by solving the optimization problem of 2.16). Indeed, projection for LLP is simply done by linear combination of weighted original samples and therefore (as weights are normalized to sum to 1), the projection for LLP can be written with the expression of 2.17 provided that for all Y, $K(X_i, Y)$ is found *via* the optimization of 2.16 (K is not a proper kernel, but for convenience purposes we will often denote it as such). We separately introduced LLP on itself as it is a much more intuitive and simple method that the more general KMP one, that does not need NWKR for projection.

As KMP can be used with any kernel (provided we can extend it to any

new sample), we will denote it as **KMP**, while specifying the kernel in used (diffusion or geodesic).

2.4.3 Conclusion

In this section, we presented two original methods for manifold projection that were designed during this thesis. Both have roots in the methods presented earlier in 2.3 and are based on the Nadaraya-Watson kernel regression of section 1.4.3.

- A first technique is **derived from the LLE algorithm** 1.2.2, but where the embedding computation is actually not required. We rather use the weights obtained from the LLE algorithm (which are usually computed in the purpose of finding points in low-dimensional space that respect those weights), and directly use them to find our projection as a local, linear combination of training samples. As it is a **locally linear projection**, we naturally refer to this method as **LLP**.
- The second one is what we call **kernel manifold projection** (**KMP**), in which we use the Gram matrix of the kernel being used, sampled over the traning set and extended to the test sample, as the **kernel matrix used in the NWKR**.

2.5 Synthesis of methods and discussion

At this point we have introduced a wide variety of methods that are all seemingly close looking, while each having their own peculiarities. They all have the common goal to perform a non-linear modelization of our embedding. But which are theoretically the closest to achieving this purpose? What is the most common denominator between these methods? What performance can be expect from a method given the performance of another one? In this section we propose to synthesize each of the presented methods, and to have an open discussion about the modelization used in our algorithms, compared to the state-of-the-art methods or to other approaches that could have been considered.

2.5.1 Method synthesis

Method	Dimension Reduction	Out-of-sample Extension	Reconstruction
GLM	None	None	Mean of the training set
PCA	Eigenvectors of the covariance matrix	Same projection as training	Analytic reconstruction
ISO	Eigenvectors of the geodesic Gram matrix	Nyström extension	NWKR with gaussian kernel computed over the embedding distances
KMP (Locally Linear): LLP KMP (DM): KMP_{DM} KMP (Geodesic): KMP_{ISO}	None	Kernel extension	NWKR with weights computed using the extended kernel
DM	Eigenvectors of the diffusion Gram matrix	Nyström extension	Reconstruction as a convex sum of training samples with weights obtained by optimization in the embedding
ISOPTIM	Classic Isomap one	Optimization in the embedding	NWKR using the gaussian kernel computed on the result of the optimization and the embedding

Table 2.1: Summary of all the presented methods for kernel manifold projection.

A few remarks over the summary presented in table 2.1:

- The GLM is obviously the simplest of methods.
- PCA is the only one (with GLM) to be fully analytical.
- All our non linear methods have quite similar reconstruction methods.

2.5.2 Discussion

As an observant reader will have noticed at this point, **all** our methods are actually looking for a (manifold) projection in the form of a linear combination of training samples (albeit with weights that are computed in various non-linear fashions). This is obviously a **highly constrained** modelization, which raises the legitimate question:

Problematic: Is our modelization a true non-linear one, able to capture all the behaviour of a non-linear subspace of the euclidean one, when our reconstruction is based on a pseudo-linear modelization?

The answer to this question is rather complex, and possibly at the heart of all the work presented in this manuscript. We cannot provide a full answer at this stage (if we dare say that we can provide one at all!), but we can provide the elements of reflection that have guided us to this kind of modelization:

- First, we should state that while not being fully linear, all (or most) of our methods are at least much more non-linear that the likes of GLM and PCA: we are not really close to a fully linear reconstruction as our weights are computed either by optimization of truly non-linear distances, or only expressed on neighbourhoods of the test sample (therefore using the locally linear hypothesis of our manifold). In simple case, this is easily enough to model a non-linear set, and outperform a linear one.
- Several of our methods rely on optimization to obtain a preimage, which could be a key to provide a projection with a fully non-linear modelization. Ours plainly rely on the linear combination of training samples, but one could provide a projection by using an optimization in the original space. This is evidently much harder than it sounds, and relies on finding a good cost function (preferably differentiable) that respects the manifold. This can be done depending on the data being used: [80] have presented one in the context of shape manifolds, and [82, 83] for manifolds of medical images, but none were adapted to our problem.
- Performing a fully non-linear reconstruction is actually **in our opinion**, a much harder task than performing a non-linear dimension reduction. We base this on the fact that while the dimension reduction task (roughly) consists in simplifying the information from a high-dimensional

space to a low-dimensional one (like a compression algorithm), the reconstruction must try to reconstruct all the details from an image and all the non-linearities from a very complex dataset, starting from what is only a simplified version of it. One could obviously try to use fully non-linear methods of regression to perform the reconstruction, as we already stated in section 1.4 (and as we tried), but it is our belief that a completely non-linear regression method would not perform any better on such a task of regression, and that performance would overall be the same (see figure 2.5 for more on this topic).

Figure 2.5 represents the problem of trying to apply our paradigm of projection to an insufficiently sampled manifold. We first consider a manifold in a high dimensional space where a "curved" portion of the manifold (representing the details we mentioned during the discussion) has not been sampled by our training set. The first line presents 3 point on a "curved" manifold: X_1, Y_2, X_3, where only X_1 and X_3 belongs to the training set (Y_2 belongs to the manifold but has not been sampled). The second line of the figure represents the embedding that can be obtained with a non-linear dimension reduction tool. The last line shows what the reconstructed points from the embedding with any non-linear method of reconstruction should look like, without further modelization of the manifold.

Trying to find the projection of any point Y_2 close to this curve with our paradigm and without any additional modelization is hopeless: its out-of-sample projection, based on the distances to closest neighbours (X_1 and X_3), even if perfectly correct (in our example, right at the middle of its two closest neighbours embedding, as the original space is in the *geodesic mean* of the two high dimensional samples) is not enough information for any method to be able to reconstruct the curved part that as not been sampled any better than what our own methods will do (essentially a straight line between the two neighbours).

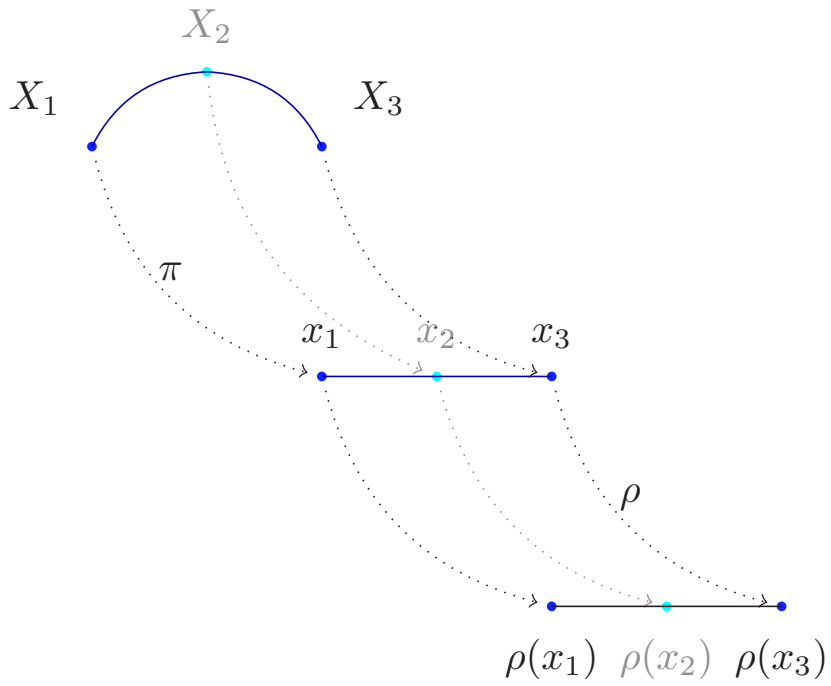

Figure 2.5: The sampling problem in manifolds: providing a pertinent reconstruction for a test point located in a previously unsampled area of the manifold is extremely challenging. X_1, Y_2 and X_3 belong to the same manifold represented in the first line, but Y_2 is not in the training set. The dimension reduction is found for each of them, and a reconstruction is made based on these dimension reduction.

Overall, it seems to us that this modelization is a good compromise: it should be simple enough to "interpolate" our manifold between training samples, perhaps while loosing some details of the manifold structure if we have not been able to sample it in "high frequency areas", but never as much as a linear modelization (provided that the dataset is indeed non-linear).

Remark: While our non-linear methods have some form of proximity with PCA by having a reconstruction that is a linear combination of training samples, the PCA reconstruction itself is not based on training samples, but rather on components that are extracted from the covariance matrix of the dataset, thus rising the question of which is better to be used for our images reconstruction.

As we already raised in section 1.3, none of the methods for manifold projection we presented in this chapter are inherently robust to outliers introduced by a pathology. Therefore, the following section is dedicated to robust versions of the algorithms we introduced.

2.6 Robust Extensions

Contents for this section

In this section, we will focus on adding a "robustness layer" to our manifold projection methods. All of the methods we presented are perfectly able to perform as intended over samples that belong to the manifold of normal images, but that are not part of the training set, such as the validation samples used for statistical testing (see section 2.1). Indeed the projection in this case is quite an easy one, as projecting a point onto a manifold that belongs to said manifold *should* amount to do nothing. As we only sampled the manifold of normal images, with relatively few samples (compared to the dimensionality of the manifold), we do not expect the projection of normal samples to be this easy but at least to be manageable by our algorithms.

2.6.1 The Need for Robustness: the Era of Fake Relationships

The task is however much more difficult on samples presenting anomalies linked to their pathology, as introducing anomalies will inevitably result in a sample that is further and further away from the manifold as the intensity of the anomaly increases. This notion of remoteness is especially important with our multivariate approach and the algorithms that we developed upon it (as opposed to the GLM, for instance), as we always rely on *some sort of proximity* to perform our projection. Isomap, LLE, PCA and the diffusion maps all rely on the L_2 distance and/or the associated closest neighbours to consider the relationship between a test sample and the ones of the training set. If our test sample were to be altered by the anomalies introduced with a pathology, these L_2 distances (and therefore these L_2 neighbourhoods) should in turn be greatly altered, therefore introducing fake relationships with training samples and providing a final result that is not what we could expect (see figure 2.6 and 2.7 for an analysis of the alteration of distances and neighbours introduced by an anomaly). These relationship are the heart of our methods: essentially they will assume for them to be correct, while attempting to recover a point that is coherent with the sampled manifold and respecting these relationships. The more relationships are changed, the further the resulting projection will be from the normal version of our test sample. This phenomenon is peculiar to us, as it is widely unintuitive: a highly modified image should be easier to

detect than a lightly altered one, but in our case, our projection (and therefore our anomaly detection) will be better for the lightly altered one. Univariate models such as the GLM are not as easily disturbed by anomalies, on the contrary: the further a voxel intensity is to the mean intensity of the training voxels, the better it will be detected.

In figure 2.6, we used a set of MRI jacobian images (see section 3.3 for further introduction on this dataset) as a test case. We randomly selected one sample Y amongst all the normal ones (about 1000), and computed the L_2 distances to its 20 closest neighbours. Then, we introduced an anomaly by adding a multiple of the voxel-wise variance vector (computed on the 1000 healthy MRI scans) in the ventricle area of the brain (representing between 5% and 10% of the brain). By controlling a multiplication factor of the variance vector, we control the norm of the perturbation $P = \alpha\sigma$ we introduce. We then plotted the effect on the original L_2 distances between Y and its neighbours against the distances between $Y + P$ and the closest neighbours of Y: the lighter a data point appears, the greater the norm of the perturbation is. We can see the large effect of the perturbation P over the original distances. Nearly all are modified (and not in the same way, as some increases whereas others increases), thus introducing fake relationships between our samples that will inevitably lead our algorithms to provide mistaken projections.

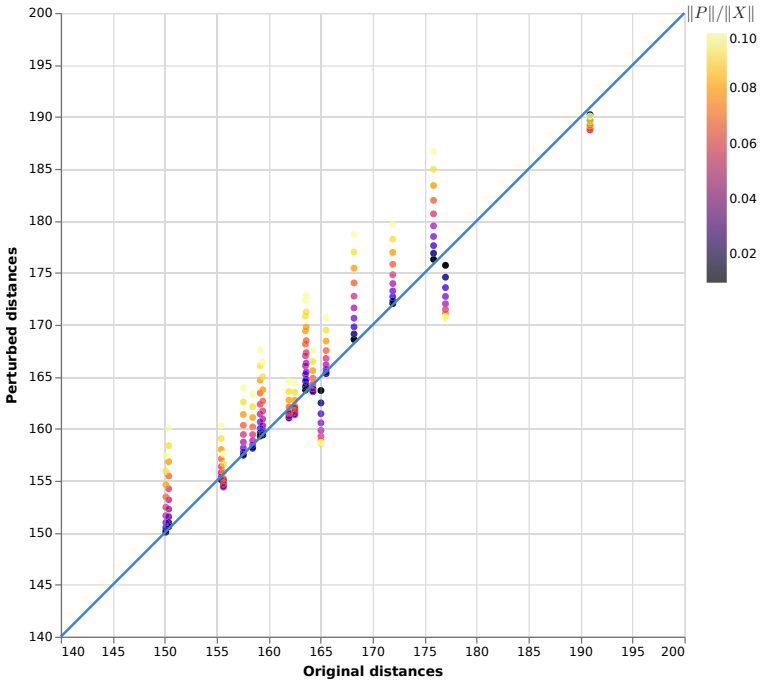

Figure 2.6: Scatter plot representing the L_2 relationships between a normal samples and its 20 closest neighbours in the training set, along with the altered counterparts (the lighter the point, the more it has been altered), depending on the norm of the perturbation introduced. A vastly altered sample will have L_2 distances to the training set widely different than the original image.

In a similar fashion, in figure 2.7, we looked at not only one test subject, but a hundred. For each one, we computed the "original" 20 closest neighbours, introduced a perturbation just as before, and recomputed the set of neighbours for each test sample. This figure presents the common fraction of neighbours between our test samples and the original, unaltered counterparts. What we observe is that this fraction of common neighbours will (globally) rapidly decrease as the norm of the perturbation increases, and thus our algorithms will use incorrect neighbours to project our test samples.

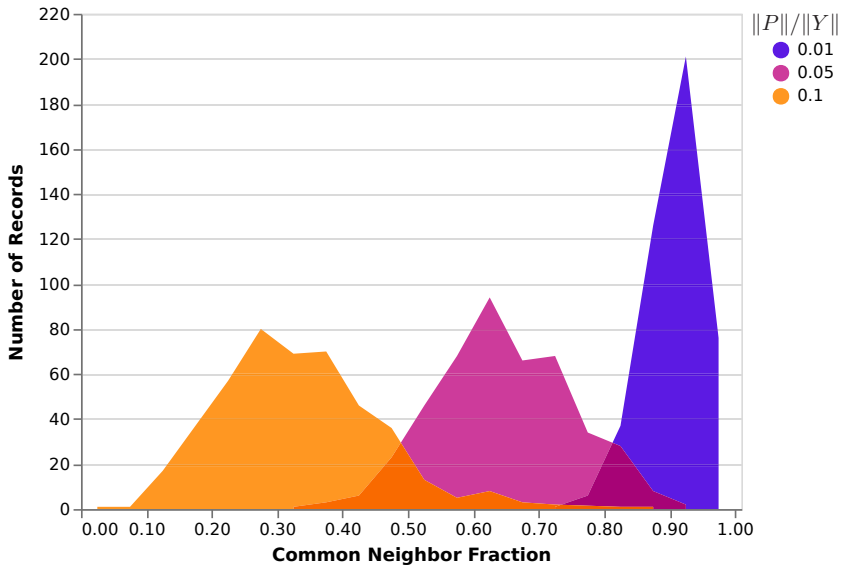

Figure 2.7: Multiple histograms representing the common fraction of neighbours (inside the training set) between a normal sample and its altered counterparts depending on the norm of the perturbation introduced. A vastly altered sample will have nearly no neighbours in common with the original image.

What we take from these experiments is that we cannot stress enough the **need for our algorithms to be robust**, if we want to have the same kind of performances between highly suspect samples and less suspect ones.

Let us now take a closer look at the robust version of PCA, which is at the origins of our algorithm that provides a robust version for any of our methods.

2.6.2 Robust PCA

The PCA algorithm has dealt with its inherent non-robustness (see section 1.3.1) by transforming its L_2 distances into a robust version with the help of a robust function f. Such a function is generally designed to present a quadratic behaviour when the Z-score is small (in order to ensure we obtain the same performance over normal subjects as we would without the robust overlay), and to rapidly decrease to 0 when the Z-score magnitude is high (so that the influence of a highly suspect voxel is null).

In [1, 35], the quadratic cost function associated with PCA is replaced with a robust cost function that reduces the weight given to outlier voxels in the image. The dimension reduction is still acquired *via* maximum likelihood estimation, but this time the solution of the optimization problem (roughly presented in 2.18), has no analytic solution, and is non convex. One can however attempt to find an iterative solution thanks to an iterative reweighted least-square (IRLS) [84, 85, 86] optimization scheme. IRLS offer a local solution to non-convex problems (under conditions over our optimization function that are validated in our case).

$$\tilde{\pi}_{robust}(Y_n) = \arg\max_{x \in \mathbb{R}^m} \exp\left(-\frac{1}{2}\sum_s f\left(\frac{(Y_n)_s - \rho_{PCA}(x)_s}{\sigma_{LS}}\right)\right) \tag{2.18}$$

in which σ_{LS}^2 is the variance computed at this voxel for the classic least-squares residuals obtained with the unchanged PCA method, which is passed as a parameter of function f. σ_{LS} is computed on the validation set V, but is **not to be confused** with the variance we will obtain with our robust method and use further on to compute Z-scores for our test samples.

With IRLS, a weighting scheme is introduced to put in effect how the robust function is reducing the effect of outlier voxel values: a voxel which has a suspect value (given the training set values for this voxel), will not be dealt with as PCA might do. In the classical PCA algorithm, each voxel has the same contributing power, and the distance function of PCA being quadratic, voxels with widely different values from the training set drastically contribute to our cost function. In the robust PCA algorithm 2 however, the influence of highly-irregular voxels is tuned down by the robust function, with the help of the weighting system: the suspect voxel individual cost in the global cost function is decreased until reaching 0 for extremely abnormal values. This weighting problem intuitively calls for an iterative method in which we will aternate between computing weights and residuals using these weights. The IRLS optimization scheme therefore transforms the original problem of solving equation 2.18 into solving a weighted least-square problems at each iteration:

$$\tilde{\pi}(Y) = \arg\min_y \|B(Y - \overline{X} - Wy)\|_2^2 \tag{2.19}$$

These optimization subproblems have an analytic solution quite close the least-squares one, as can be seen from step 4 of algorithm 2.

A popular robust function (and consequently the one we will be using) is Tukey's bisquare function:

$$f(x) = \begin{cases} (1 - (\frac{x}{h})^2)^2 & \text{if } x \leq h \\ 0 & \text{otherwise} \end{cases} \tag{2.20}$$

where h is a scaling parameter typically [87] chosen to be $h = 4.685$ for 95% efficiency when the samples are normal.

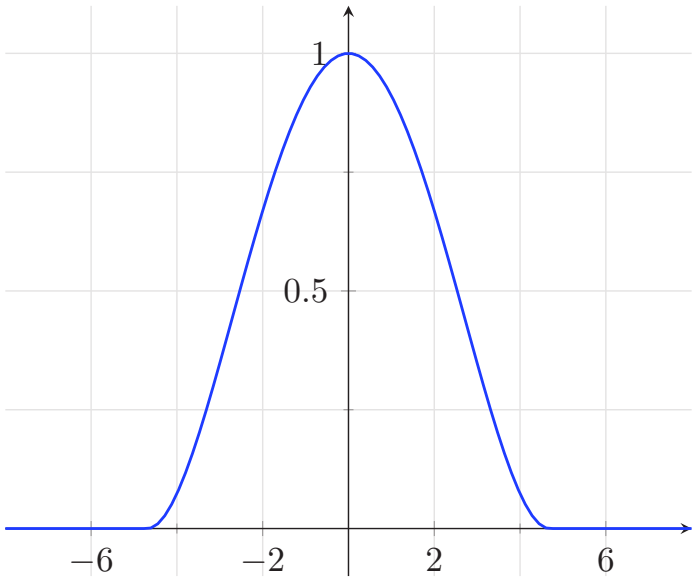

Figure 2.8: Tukey's bisquare robust function with $h = 4.685$.

Algorithm 2 ARTUR algorithm for IRLS

Require: X dataset, Y test, f robust function, ϵ stopping criterion, σ_{LS}

$B^{(0)} = I_d$ ▷ Initialize with PCA solution
$k = 0$
while $\|B^{(k)} - B^{(k-1)}\| > \epsilon$ **do**
$k = k + 1$
$\tilde{\pi}^{(k)}(Y) = (W^T B^{(k-1)} W)^{-1} W^T B^{(k-1)}(Y - \overline{X})$ ▷ Compute robust DR of Y
$r^{(k)} = Y - \overline{X} - W\tilde{\pi}^{(k)}(Y)$ ▷ Compute RLS residual for Y with non robust reconstruction
$b_s^{(k)} = f'(\frac{R_s^{(k)}}{(\sigma_{LS})_s}) / \frac{2R_s^{(k)}}{(\sigma_{LS})_s}$ ▷ Compute the weights associated to each voxel
$B^{(k)} = \begin{pmatrix} b_0^{(k)} & & \\ & \ddots & \\ & & b_s^{(k)} \end{pmatrix}$
Return last residual $R^{(k)} = Y - \overline{X} - \rho_{PCA}((\tilde{\pi})^{(k)}(Y))$

This iteration scheme provides us with an algorithm for projection that we use, as described in our paradigm, on the validation set to compute the variance of residuals σ_V with a robust version of PCA. Finally we compute a Z-score for test samples with σ_V as in equation 2.1.

2.6.3 Robust non-linear projection

2.6.4 ISOPTIM extension

The closest formulation to PCA is the one of ISOPTIM, and is the only one of our methods is actually expressed as a least-square optimization like PCA. We can transform it in a similar way to equation 2.18 to produce a M-estimation for ISOPTIM (see equation 2.21).

$$\tilde{\pi}_{robust}(Y) = \arg\max_{y\in\mathbb{R}^m} \exp\left(-\frac{1}{2}\sum_s f\left(\frac{Y_s - \rho_{NWKR}(y)_s}{\sigma}\right)\right) \tag{2.21}$$

We can define just as well an IRLS scheme to provide a solution for 2.21 as we did for PCA. The "transformed" problem of equation 2.21 by IRLS is now to find the solution to the weighted least-squares problem of algorithm 3 (step 5). Unfortunately, while in the case of PCA the solution of this problem was analytic, this is not he case here, as our least-square problem is this time non-linear. Therefore, each step of IRLS for ISOPTIM leads to a new optimization problem that we must solve with conventional gradient methods (unlike for RPCA where it is solved analytically) similar to the original non-robust one of ISOPTIM. This is however only a computation time concern for our problem.

Just as for the RPCA IRLS, we first need to compute the variance of validation residuals using the non robust version of ISOPTIM (that we will denote σ_{NR} in the description of the IRLS algorithm for convenience reasons). Then we start an iterative algorithm using weights obtained in the same manner as for RPCA, in which the residuals obtained with the previous weights are used to compute the new weights. The following algorithm 3 details the computation steps of the IRLS scheme associated to our robust ISOPTIM.

Algorithm 3 Algorithm for robust ISOPTIM manifold projection

Require: $X, V, \epsilon, Y, f, \sigma_{NR}$

1: $R^{(0)}(Y) = Y - \mu(Y)$ ▷ Initialize with ISOPTIM solution

2: $k = 0$

3: **while** $\|B^{(k)} - B^{(k-1)}\| > \epsilon$ **do**

4: $b_s^{(k)} = f'(\frac{R_s^{(k)}(Y)}{(\sigma_{NR})_s})/\frac{2R_s^{(k)}(Y)}{(\sigma_{NR})_s}$ ▷ Compute weights for each voxel

5: $\tilde{\pi}_B^{(k)}(Y) = \arg\min_y \|BY - \frac{\sum_i K(x_i,y)BX_i}{\sum_i K(x_i,y)}\|^2$ ▷ Compute the robust Out-of-sample extension

6: $R^{(k)}(Y) = Y - \rho(\tilde{\pi}_B^{(k)}(Y))$ ▷ Compute the residual using the extended, weighted kernel

$k = k + 1$

7: **Return** last residual $R^{(k)}(Y)$

Remark: The weighted extension of the the kernel is the same as the ISO one, and will be given in the next section.

2.6.5 Extensions for Kernel Methods

It is a bit more complicated to provide a robust algorithm for other non-linear methods that we developed. As they are not linked to a least-square optimization problem, we actually have to try to transpose the IRLS concept to our methods without the same guarantees (of convergence, albeit a local one) as for PCA or ISOPTIM. As in an IRLS scheme (although this is not one here, as we are not minimizing any maximum likelihood function), we want our projection μ to depend on weights that would eliminate the influence of suspect voxels according to the training set. Therefore we performed a similar approach to IRLS in order to "robustify" our projection, by using an iterative scheme of projection using the result of the last anomaly detection as weights for the next one, in order to reduce the importance of such voxels. This weighting system had to be introduced at the out-of-sample extension level, after we computed the embedding for the training set (as no training sample is supposed to be anomalous, by definition), and before the computation of any distances or neighbourhoods between our test sample and the training set, as figs. 2.6 and 2.7 taught us.

The first step of any non-linear robust algorithm, as for PCA or ISOPTIM, is to compute the residuals for the validation set V with an unchanged, non-robust, method and to compute the voxel-wise variance of residuals across all subjects. Then, again, we use this variance as a parameter of a robust function as a part of our robust algorithm of manifold projection. A first non robust projection of our test sample Y is performed, to initialize the weights. Then we iterate the computation of the weights and the projection of Y until convergence, where the algorithm returns the latest projection of Y, obtained with the optimal weights. What differs from algorithm to algorithm, is the manner in which we performed the weighting of our voxels. Algorithms for robust projection with the ISO 2.3.1 projection and the KMP method 2.4.2 are given respectively in 4 and 5 to shed light on their specific processes of weighting.

Similarly to ISOPTIM, both algorithms rely on the extension of their kernel once the weights have been computed, *i.e.* the new kernel should take the weights into account. ISO presents the additional constraint that we must recompute the embedding using this weighted kernel, as to avoid discrepancies between distances inside the training set and distances from training set to test point.

To provide robust extensions for non-linear methods, we must weight the kernels we presented and extended in section 1.2 using weighted distances instead of classic L_2 distances.

Algorithm 4 Algorithm for robust ISO manifold projection

Require: $X, V, \epsilon, Y, f, \sigma_{ISO}$

$R^{(0)}(Y) = Y - \mu(Y)$ ▷ Initialize with ISO solution
$k = 0$

while $\|B^{(k)} - B^{(k-1)}\| > \epsilon$ **do**
$k = k + 1$
  $b_s^{(k)} = f'(\frac{R^{(k)}(Y)}{(\sigma_{ISO})_s})/\frac{2R_s^{(k)}(Y)}{(\sigma_{ISO})_s}$ ▷ Compute weights for each voxel
  $\tilde{K}_B(Y, X_i)$ ▷ Extend the kernel while weighting each voxel
  $y^{(k+1)} = \tilde{\pi}_W(Y)$ ▷ Apply Nyström with extended, weighted kernel
  $x^{(k)} = \pi_B(X)$ ▷ Recompute the embedding for distances consistency
Return last residual $R^{(k)}(Y)$

Algorithm 5 Algorithm for robust KMP manifold projection

Require: $X, V, \epsilon, Y, f, \sigma_{KMP}$

$R^{(0)}(Y) = Y - \mu(Y)$ ▷ Initialize with KMP solution
$k = 0$

while $\|B^{(k)} - B^{(k-1)}\| > \epsilon$ **do**
$k = k + 1$
  $b_s^{(k)} = f'(\frac{R_s^{(k)}(Y)}{(\sigma_{KMP})_s})/\frac{2R_s^{(k)}(Y)}{(\sigma_{KMP})_s}$ ▷ Compute weights for each voxel
  $\tilde{K}_B(Y, X_i)$ ▷ Extend the kernel while weighting each voxel
  $R^{(k)}(Y) = Y - \mu_B^{(k)}(Y)$ ▷ Compute the residual using the extended, weighted kernel
Return last residual $R^{(k)}(Y)$

This means we obtain for the geodesic kernel:

$$\tilde{K}_{B,geo}(Y, X_i) = \min_{U \in \mathcal{N}_{BX}(BY)} K_{geo}(U, X_i) + \frac{d(BY, BU)}{\overline{B}}$$

Here, we look for neighbours of Y in X using weighted L_2 distances: $\|B(X_i - Y)\|_2^2 = \sum_s b_s^2 (X_s - Y_s)^2$ (*ergo* the abuse of notation BX). For ISO, we need to recompute the embedding using this kernel, as we will further on compute distances in the embedding as part of our reconstruction scheme between training points and testing ones. For the kernel to be consistent between robust distances (used when testing an abnormal sample) and classic L_2 distance (computed between every control samples), we need to normalize the robust distances, in order for them to have globally the same amplitude as non-robust one. We performed this by normalizing using the weights mean. The Nyström out-of-sample extension is then computed using this kernel instead of the usual

one, and the reconstruction is done classically with NWKR (as the point in the embedding should present no trace of anomaly).

For the neighbourhood kernel:

$$\tilde{K}_{B,LLE}(Y, X_i) = w(BY, BX_i)$$

We simply recompute neighbours of Y using weighted distances, and then reckon the weights to be the ones of weighted version of Y and its neighbours in X.

Finally, for the diffusion kernel:

$$\tilde{K}_{B,DM}(Y, X_i) = \frac{k(BY, BX_i)}{\sqrt{\mathbb{E}_j[k(BY, BX_j)]\mathbb{E}_k[k(X_i, X_k)]}}$$

As for ISO, a normalization of robust distances is actually needed before computation of the robust kernel $k(BY, BX_j)$. This is done by dividing beforehand the L_2 distances by the weights mean, just as for ISO.

Note: The weighted diffusion kernel will only be used in the KMP setting, as due to computational issues, no robust DM method could be developed.

2.6.6 Partial Conclusion

In this section, we introduced **original robust extensions** of our methods. We presented a **cost function for PCA** from previous works able to incorporate robust functions that decreases the importance of abnormal voxels. This new cost function being **non-convex** and having no analytical solution, we presented a class of optimization algorithms called iterative reweighted least-squares, that provides an **iterative, weighting scheme** to solve our optimization problem. The non-linear method ISOPTIM, being expressed as a least-square problem, can also be dealt with an IRLS algorithm. Finally, we adapted this class of algorithms to our non-linear methods, with the additional difficulty of having no cost function to guide us.

Chapter 3

Getting a Better Understanding of the Problem

This Chapter contains:

In this chapter we will attempt to highlight the strengths and weaknesses of our methods, by using them to perform a voxel-wise anomaly detection in a subject vs group setting. Our analysis will obviously also try to emphasize the added value of the robust layer that we superimposed to the original methods, by naturally comparing each method with its robust version. While we would like to find a clear and consistent hierarchy of methods across multiple datasets, we must understand that each dataset has its own geometric specificities (linear/non-linear, high/low latent dimension, few/many samples, etc.) that will naturally impact the results of each method in their own way. Therefore, our main objective is rather to obtain a better understanding of how the methods work and what makes them fail.

As a start we will present results over multiple synthetic datasets from the simplest modelization of a half-sphere (section 3.1) and increasingly complexifying the underlying model of our data, first into non-linearly sampled images (section 3.2) to finally reach a modelization (section 3.3) consistent with what we have in our real datasets (that will be dealt in a second time in chapter 4). The synthetic datasets results will thus provide us with a deep analysis of both our methods and paradigm, and a sense of what to expect on real ones. The synthetic dataset analysis is also made necessary by the fact that real datasets do not come with a ground truth that allows for a quantitative analysis of the results.

First, let us introduce the synthetic datasets by starting with a simple, geometric one.

3.1 A Geometric Dataset

Contents for this section

3.1.1 The Dataset

As our main hypothesis 1.1.1 is that real world data is not sampled from linear, euclidean spaces but rather from non-linear (yet smooth) manifolds (see appendix 5.1.3), our synthetic datasets ought to present such properties as we wish to test our methods on challenges inspired by "real world" data. The simplest approach in creating a controllable, intuitive, non-linear data set is to start from a familiar one with a *very* small intrinsic dimensionality, from which we can easily sample. Such a dataset can be obtained by embedding an already non-linear subset of $\mathbb{R}^m$ (with m "very small") into $\mathbb{R}^d$ (with d as large as we want). This embedding is done in the easiest form, by using a linear application to get from the low dimension dataset to a high dimensional one. This modelization is therefore consistent with the assumption we made: the high-dimensional dataset is sampled from a non-linear dataset (composing a non-linear dataset with a linear function does not make it linear) with a very small intrinsic dimensionality and with an *equivalent smoothness* to the one of the original, low-dimensional dataset.

With this idea in mind, we consider an half-sphere dataset: we sample our low-dimensional dataset on a semi-sphere (as to avoid closing the sphere manifold on itself), and embed it in on *relatively* high dimension using the previously described manner. We use a sphere, as it is a quite simple object to mathematically describe and to sample uniformly (according to the sphere surface). Obviously the sphere is commonly used to refer to the 3D representation of the word, but in our case we will use the more general definition of a m dimensional sphere for our test purposes: $\mathcal{S}^m = \{M \in \mathbb{R}^m \ / \ \|M\|_2 = 1\}$. This definition allows us to test our methods in different settings with various intrinsic dimensionality. One beautiful appeal of using a spherical dataset is that, by using the particular subclass of linear applications called orthonormal applications to embed our dataset in high dimension, the geometrical property of the sphere (unit norm) in low dimension is conserved in high dimension. That is to say that a low-dimensional sphere in low dimension, embedded in high dimension *via* an orthonormal application is still on the sphere. To complete the process of generating control samples, a zero-mean gaussian noise of variance σ^2 is added independently on each component of the points. σ is

specified so that the L_2 norm of most of the noisy samples are between 0.95 and 1.05, independently from the value of d (keep in mind that the norm of the noise-free samples is 1), thus preserving *relatively well* the topology of the manifold. In order to obtain anomalies on some of our samples with the associated ground truth, we will consider that a fraction ($\alpha\%$) of our high dimensional samples components has **no added noise**, but rather an added constant which is linked to the noise variance: $k\sigma$, with $k = 4$ in our experiments.

Remark: Obviously, this dataset will be extremely challenging for SPM, as none of its hypothesis are true in this case. However, while the dataset in itself is non-linear, the projection method is linear (and furthermore orthogonal). Therefore, PCA *should* be able to find the correct mapping (up to a rotation) between low and high dimension, although without "untangling" the dataset in the low-dimensional space.

We will start by designing a *reference* test case which we will use as a point of comparison while changing each of the test parameters (low dimension: m, high dimension: d, number of training samples: N_s, percentage of abnormal area: α) one by one in order to get a sense of their influence over our methods. The reference case is meant to be an *easy* one and therefore will have a low dimensionality with a high number of samples, while the anomaly is not affecting a large part of our test samples. The reference case will be denoted T_{ref} and its parameters are the following ones:

$$\begin{cases} N_s = 10000 \\ \alpha = 5 \\ m = 3 \\ d = 100 \end{cases} \tag{3.1}$$

Based on this test, we define five experiments, where parameters of the reference case parameters are changed one by one, with the rest of the parameters identical to the reference ones: large modifications ($\alpha = 30$), fewer number of samples ($N_s = 2000$) greater intrinsic dimensionality ($m = 20$), large ambient dimensionality ($d = 1000$).

To evaluate the performance of the methods over each of these experiments, we will obviously use the Z scores we defined in section 2.1 but we will also look into the mean-squared errors (MSE) of reconstruction of our algorithms. Of course, anomaly detection and performance in reconstruction are closely linked, and indeed one can intuitively think that in order to find the most subtle anomalies inside a test sample, we need to be able to reconstruct validation samples with an accuracy similar to the amplitude of the anomaly; therefore our aim with our methods of projection *should* be to minimize errors of reconstruction on validation samples. However by doing so we run into the issue of too well reconstruct test samples (and therefore being counterproductive in our anomaly detection task), by *overfitting* our dataset in a way and having

such a complex model that it is able to reconstruct anomalies by extrapolating over the training set variabilities. Therefore we need to find a balance where our methods perform well on normal areas and *bad* on abnormal ones (in the sense that they are not able to reconstruct them). This leads us to have a look on one side at Receiver Operating Characteristic (ROC) curves where the curve is obtained by thresholding our Z scores with multiple values spanning their complete range in order to obtain true (TPR) and false (FPR) positive rates that are plotted one against the other. In ROC curves, we are particularly interested in the part of the curve where the FPR is low (at most 10^{-2}), as it is of greatest concern in medical applications. On the other side, we will analyse the MSE of our reconstruction methods on both normal samples and abnormal ones (in both the unchanged area and the one where we introduced an anomaly). In a perfect setting, we expect our methods to have the same kind of MSE over the normal samples and the part of abnormal samples where no perturbation has been introduced.

While the size of the training set (and therefore the validation set) may vary depending on the experiment, we will always keep the same number (500) of test samples, split in half between "normal" test samples that have been sampled just as training ones and that have the same noise, and abnormal ones created as described in section 3.1.1. For the sake of clarity, we will refer further on to the following acronyms for their respective area of interest.

- **N**: The first half (250) of the test samples (all components), composed of normal samples with the same modelization as train or validation ones.
- **AN**: The unaltered components of the second half of test samples (abnormal ones).
- **AA**: The altered components of the second half of test samples (abnormal ones).

A graphical representation of the creation and split of our synthetic datasets is given in figure 3.1.

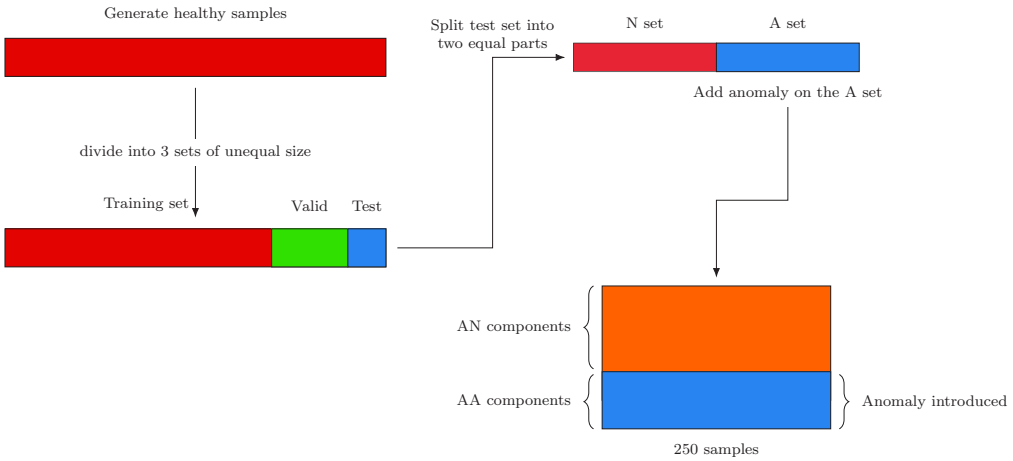

Figure 3.1: Synthetic data generation and splitting into training set and testing set.

Remark: When looking into the MSE of reconstruction in the AA area, we will compute the MSE without the anomaly added to the test sample, as to analyse whether or not a method is able to reconstruct the original test samples (*i.e.* before the anomaly was added to it) in the AA area. If we did manage to reconstruct these original test samples (the MSE in AA is small), this means our model is able to not take the anomalies we introduced in our test samples into account and extrapolate the AN area to reconstruct AA with the modelization of healthy samples. We will thus detect the anomaly as its Z score will probably be high. On the contrary, if we obtain a high MSE in AA, it means we were not able to reconstruct our original test sample and that we most likely reconstructed part or all of the anomaly we introduced, making it unlikely we detect something in this area as abnormal.

3.1.2 Reference Test Case

In all our experiments, methods requiring a low-dimension parameter have been set with the correct one m, and ones requiring a number of neighbours have been set with 20. Figure 3.2 presents a complete overview of the performance of non-robust methods over the reference test T_{ref}, in the form of ROC curves.

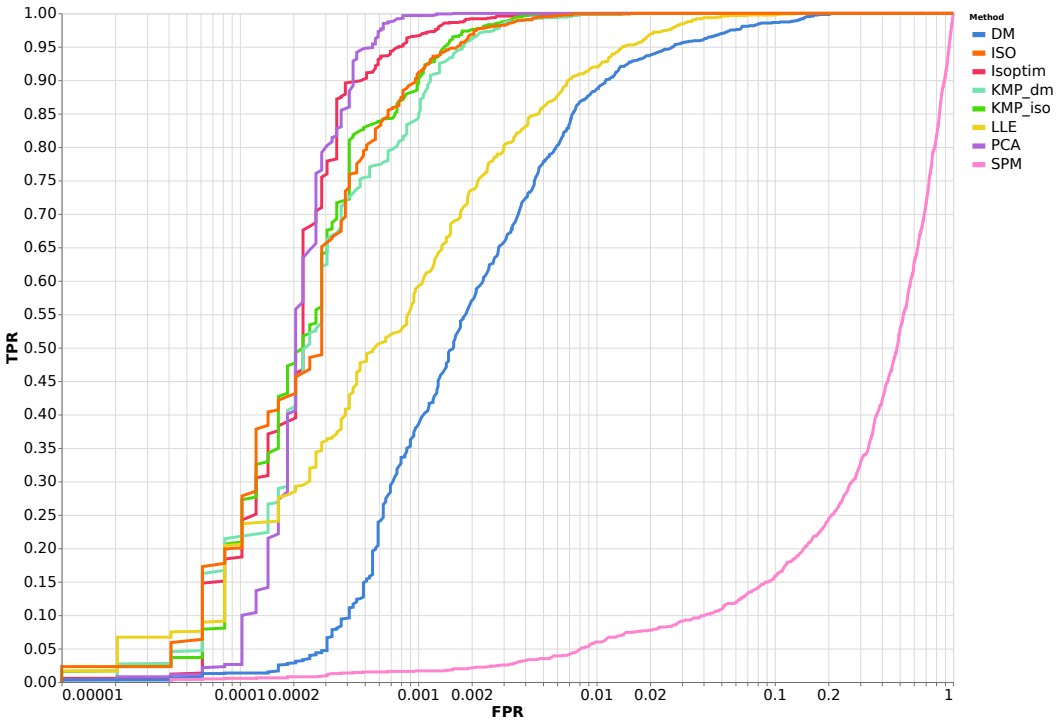

Figure 3.2: ROC curves for all the non-robust methods on the half-sphere dataset, on the reference test case ($N_s = 10^5, d = 100, m = 3, \alpha = 5$). FPR axis is in log scale.

We observe an expected behaviour for the multivariate methods, that perform really well on this reference case. Only DM has a sub-par performance on this test, that might be explained by a sub-optimal value of the diffusion parameter, that is automatically computed and might therefore not give the best results. It is noteworthy that the KMP versions of ISO and DM perform equivalently or slightly better than their original counterparts. Also noteworthy (and revealing) is the performance of PCA, which nearly bests the non-linear methods, with the correct number of components (although we did give it to the algorithm, and did not try to find it out with the help of another algorithm), *i.e.* matching the intrinsic dimension of the manifold. This is interesting, as it demonstrates that PCA can correctly model a non-linear dataset, as long as the projection itself is consistent with the PCA modelization (*i.e.* orthonormal). The poor performance of SPM was also expected, as our dataset is both multivariate and non-linear and therefore unsuited to SPM modelization.

In figure 3.3 are displayed the mean-squared error of reconstruction over our 500 test samples for non-robust methods.

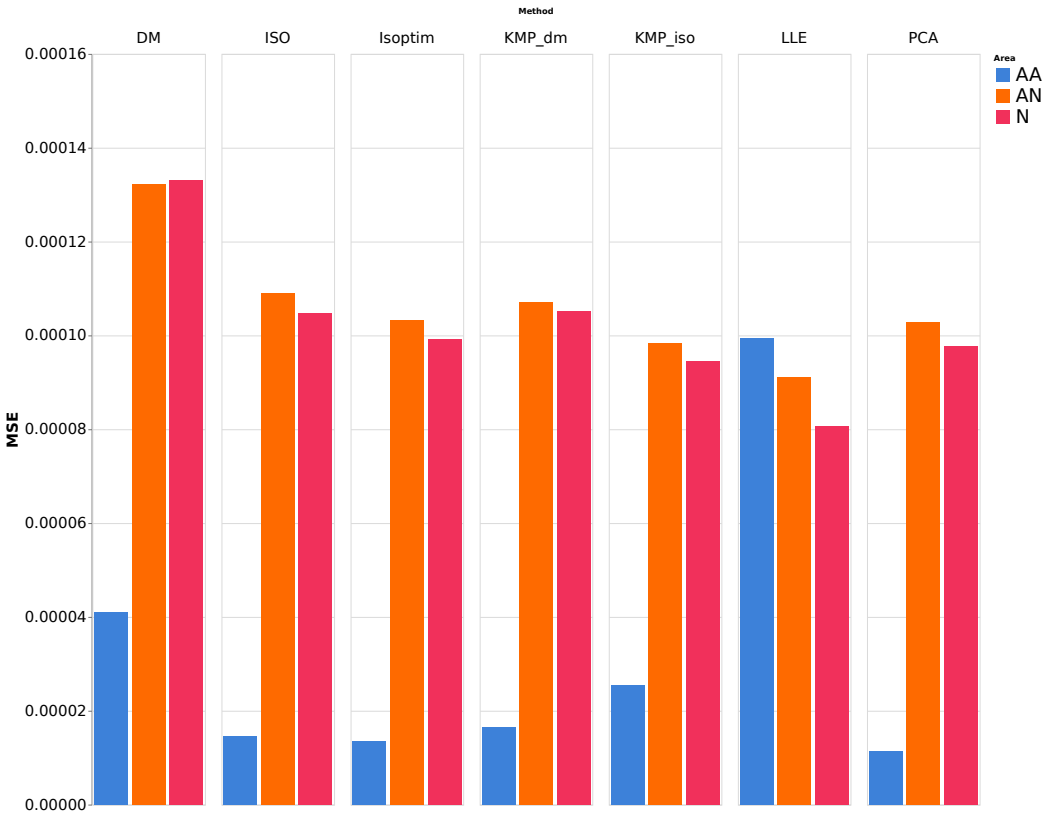

Figure 3.3: Mean-squared errors of non-robust methods over normal (N) samples, and abnormal ones (split over the normal (AN) and abnormal (AA) areas of such samples) on the reference case ($N_s = 10^5, d = 100, m = 3, \alpha = 5$).

Remark: Remember MSE on the AA area are computed without addition of the anomaly, and **without noise** on the AA area. We are thus not limited to the variance of the noise for our errors of reconstruction on these components, contrary to the AN ones. It will therefore be not surprising to observe much smaller MSE on the AA area than on the normal ones for methods that successfully recover the original sample.

It is clear that best performing (non-robust) methods are the ones that have the best reconstruction error on the AA area (*i.e.* PCA, ISO, KMP_{dm} and KMP_{iso}). The smaller the MSE in AA, the worst we will reconstruct the anomaly, and therefore the higher the magnitude of the Z score will be in these components (as the Z score *basically* compares the reconstruction error on the anomaly to the MSE of N), this combined with the fact that those methods have nearly the same performance on the AN area as in the N one (and thus few false positives) explains their strong performance on the reference case. As

for methods such as LLE, it seems apparent that the poor reconstruction (of the original, unaltered sample) provided on the AA area is the cause of their slightly weaker result: LLE is reconstructing abnormal test samples including the anomaly in some form of **overfitting** where both normal and abnormal areas are reconstructed with similar precision.

Remark: To preserve a sense of scale in the display of our MSE, the SPM ones are not plotted. Indeed, the MSEs for SPM are around 4 times larger than the worst multivariate method, and, furthermore, the difference between MSEs in AA, AN and N is also quite small for SPM.

Figure 3.4 showcases a direct comparison of each multivariate method with its robust counterpart, whenever possible (due to extreme running time over the number of samples, the DM and ISO robust version are not presented here). While not being an *extreme* improvement over the non-robust multivariate methods, the robust layer we added to the methods does not impair their results in the reference experiment. This kind of *status quo* was expected as we are dealing with anomaly only affecting 5% of the samples dimensions.

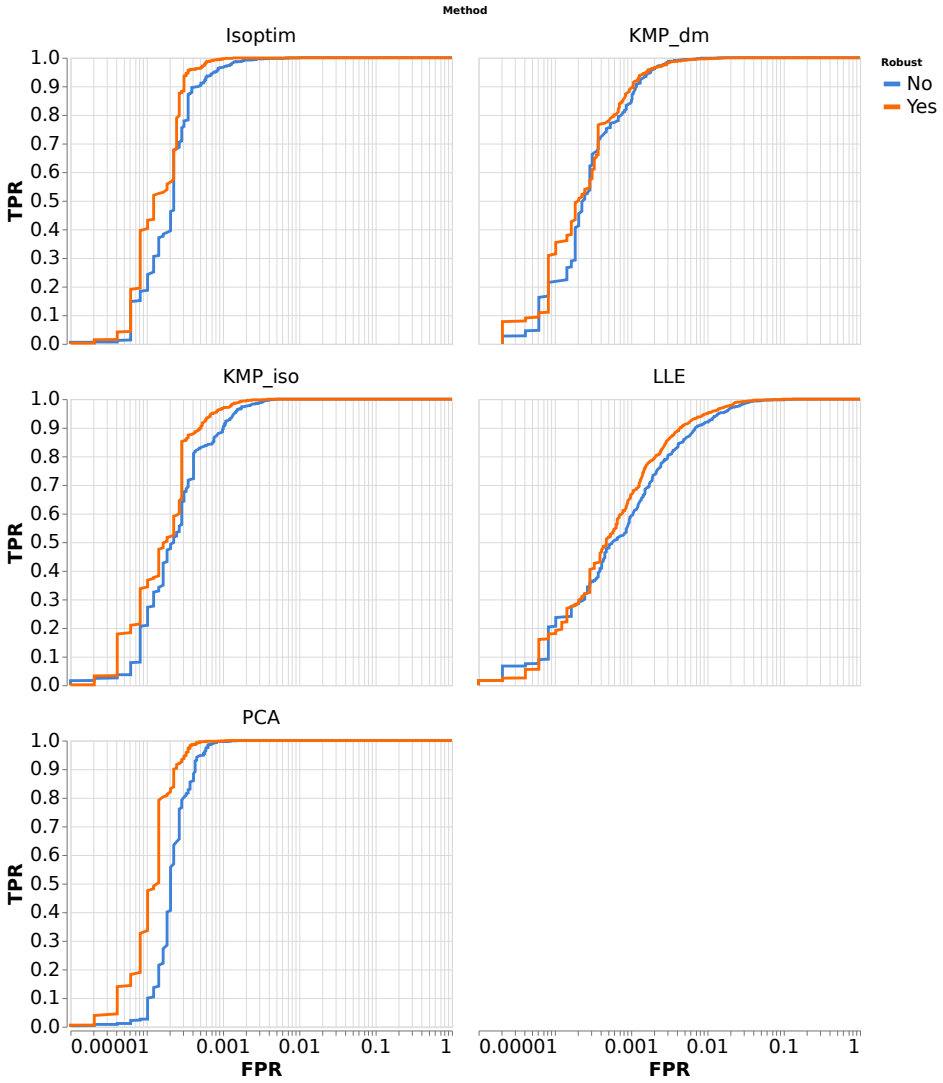

Figure 3.4: Improvement of robust versions vs non robust ones on the reference case ($N_s = 10^5, d = 100, m = 3, \alpha = 5$) shown over ROC curves. FPR axis in log scale.

The small improvements we observe can be explained by looking into the MSE results of figure 3.3 for robust methods. Robust methods (except for LLE) seem to have been able to find what part of the test samples was corrupted, and thus base their projection only on the non-corrupted areas (thanks to the weighting system of the IRLS). This is particularly noticeable on the MSEs: most of the MSE for the abnormal part decrease, suggesting that robust methods are better able to find the AA components original values for these test samples. The MSE of the normal area of altered samples (AN) also decreases, suggesting that these methods had a greater concern to correctly

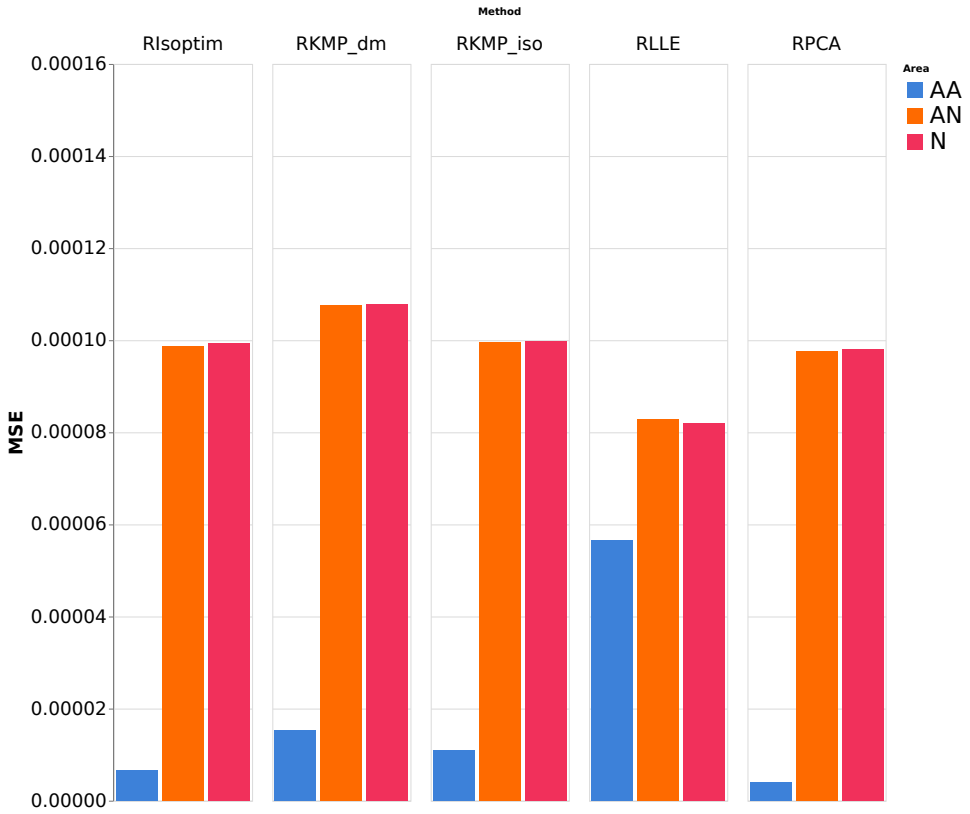

Figure 3.5: Mean-squared errors of robust methods over normal (N) samples, and abnormal ones (split over the normal (AN) and abnormal (AA) areas of such samples) on the reference case ($N_s = 10^5,\ d = 100,\ m = 3,\ \alpha = 5$).

reconstruct those area rather than the affected components (which is indeed the case given our IRLS optimization).

Remark: The seemingly *average* results of LLE are rather counter-intuitive, as we would expect the "threshold" introduced by our arbitrary set up number of neighbours to prevent LLE from overfitting. However, this dataset is typically a case where this is not enough, and where adding the convex constraints on the LLE weights would help (a lot). Indeed, while the anomaly tends to move our samples away from the sphere (as their norm is affected by the anomaly we introduced), LLE is able to provide reconstructions that have norms greatly superior or inferior to 1, by setting large weights (superior to 1) on training samples that already have norms larger than 1 (although much closer) due to the noise, and setting small weights (negative ones) on training samples with norms inferior to 1. By using a convex (or positivity) constraint on the weights, we could ensure that the norm of our reconstructed sample stays between the min and max norms of its neighbours (and thus relatively close to 1). This is obviously very specific to this dataset and as we did

not find the convex constraint to offer any concrete improvement on further experiments, we chose not to implement it further on.

Synthesis of reference case The reference test case is indeed an easy one for multivariate methods, which have strong performances on this test. PCA was one of the strongest contenders with the correct number of dimensions, as the projection is orthonormal. Non-linear methods have close results, excepted for LLE and DM which are slightly worse. All the methods where it was possible to add an IRLS layer to make them more robust to the anomaly we introduced on the test samples have seen small benefits from this robust layer. As expected, and as will be the case for the rest of the synthetic tests, SPM is completely unable to correctly modelize this kind of dataset. We now have to look at the effect of large modifications, and of the dimensionality of the problem on each of the methods performance, in order to assess which of them have the best chance of getting a strong performance on a real dataset.

3.1.3 Large Modifications Test Case

In this experiment, all parameters are set identically to the reference case, except for the size of the anomaly that is added to tests samples which increases from $\alpha = 5\%$ to $\alpha = 30\%$. As can be seen on figure 3.6, this test case is much more difficult than the reference one, which was expected as we perturbed 6 times more of the abnormal samples in this test, making it much harder for methods to perform a projection consistent with the *original* samples. All of the non-robust, multivariate methods have much lower (although *decent*) area under the curve (AUC) scores that in the reference test. PCA seems to suffer the most from its lack of inherent robustness, as it is now one of the worst multivariate method. The performance of SPM is obviously not affected as it is a univariate model.

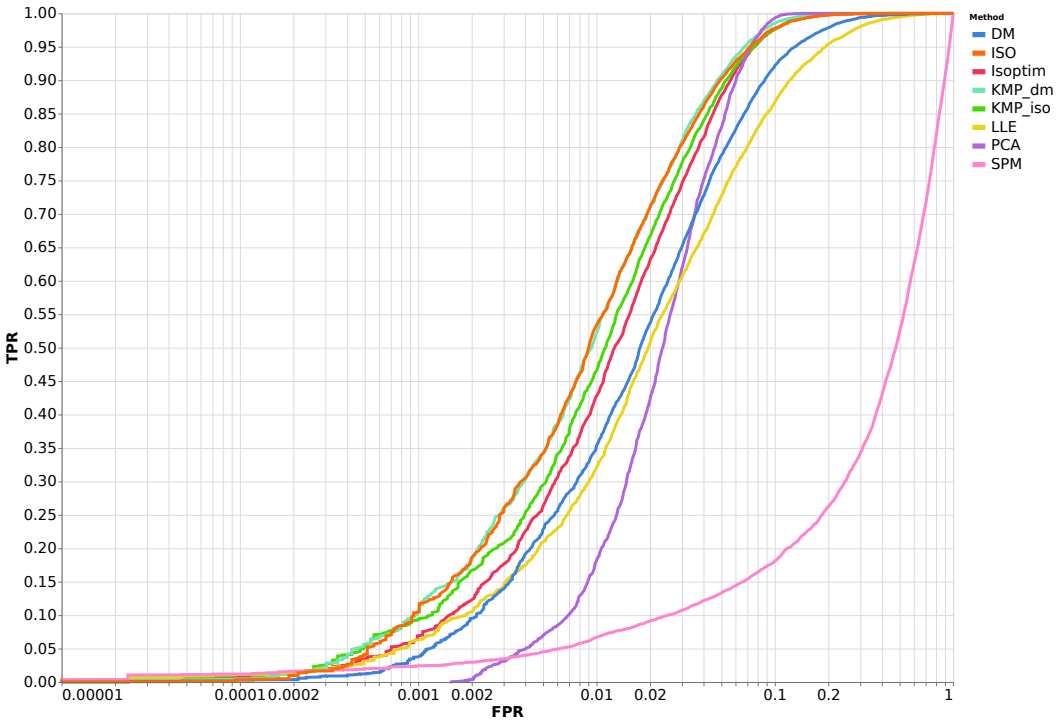

Figure 3.6: ROC curves for all the non-robust methods on the half-sphere dataset, on the large modifications test case ($\alpha = 30$). FPR axis is in log scale.

MSEs presented in figure 3.7 are also quite different from the ones of the reference case. MSEs of AA and AN are overall much larger than previously, with AN MSEs now much greater than N ones, which indicates that an anomaly of this magnitude contributes to badly reconstruct all the components of an altered sample, while this was not the case for lightly altered ones. In this experiment, the anomaly is affecting the reconstruction of non-robust multivariate models, in both altered (AA) and unaltered (AN) components by *globally* "dragging" the reconstructed sample toward the altered one.

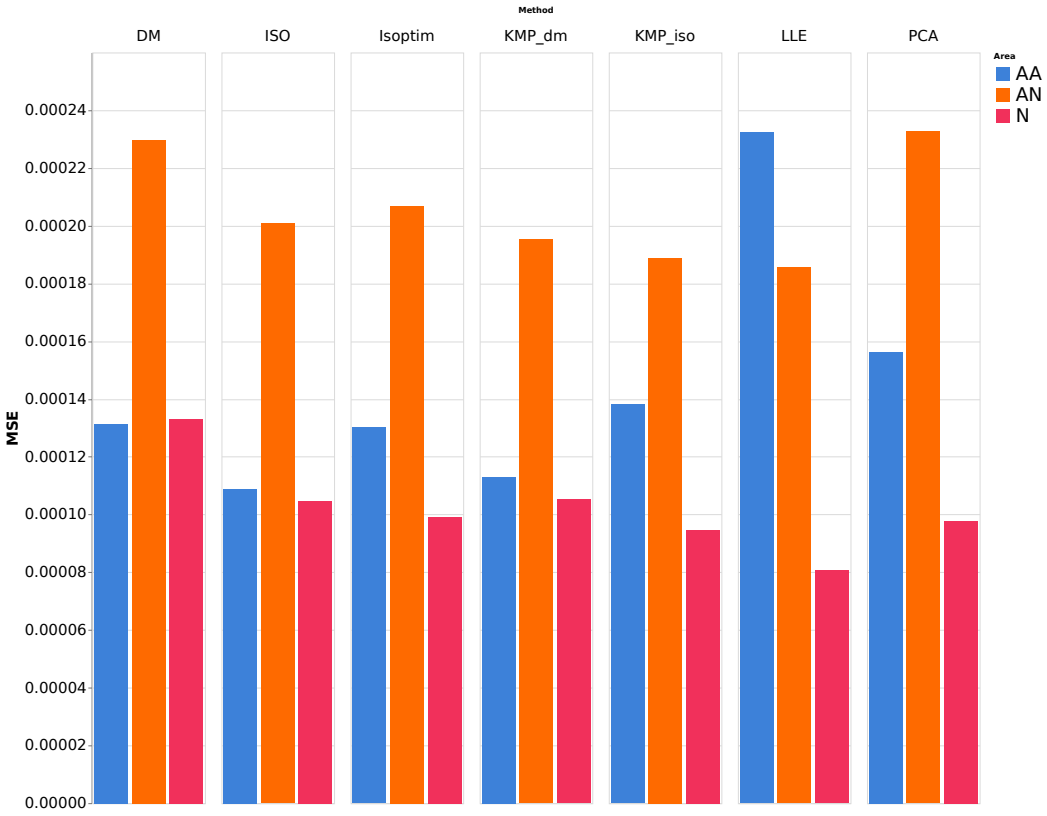

Figure 3.7: Mean-squared errors of non-robust methods over normal (N) samples, and abnormal ones (split over the normal (AN) and abnormal (AA) areas of such samples) on the large deformation case ($\alpha = 30$).

Non-robust methods have taken quite a hit and the performance (or rather, improvement) of the robust IRLS layer is critical on such a test. It is a good indicator of whether or not our methods are able to cope with large anomalies of high magnitude. Figure 3.8 presents ROC curves for each non-robust method and its associated robust version. We observe large improvements with the IRLS layer for most non-linear methods (LLE aside) and PCA, bringing us back to the standards of the reference experiment.

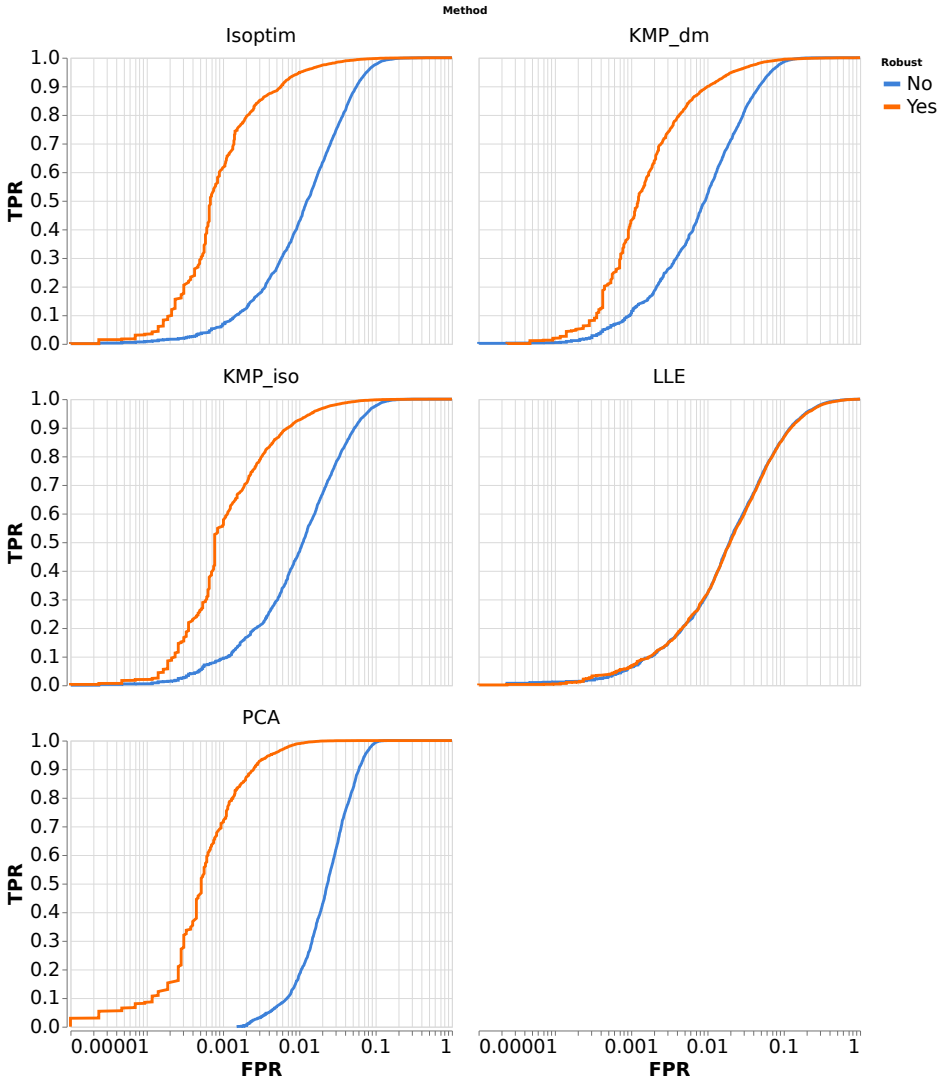

Figure 3.8: Improvement of robust versions vs non robust ones on the large modifications test case ($\alpha = 30$) shown over ROC curves. FPR axis is in log scale.

The results presented in figure 3.9 confirm what we expected: the difference between MSE in N and in AN have been mostly corrected (reducing the number of false positives), and the MSE in AA suggests the reconstructions are now much closer to the unaltered test samples (increasing the number of true positives) than before the IRLS, thus consolidating the interest for a robust layer in large modifications settings. For LLE, as suggested by the high MSE in AA prior to the IRLS, the algorithm reconstructs the anomaly nearly as well as normal areas and therefore IRLS is unable to provide a more interesting solution as LLE will reconstruct the anomaly at each step.

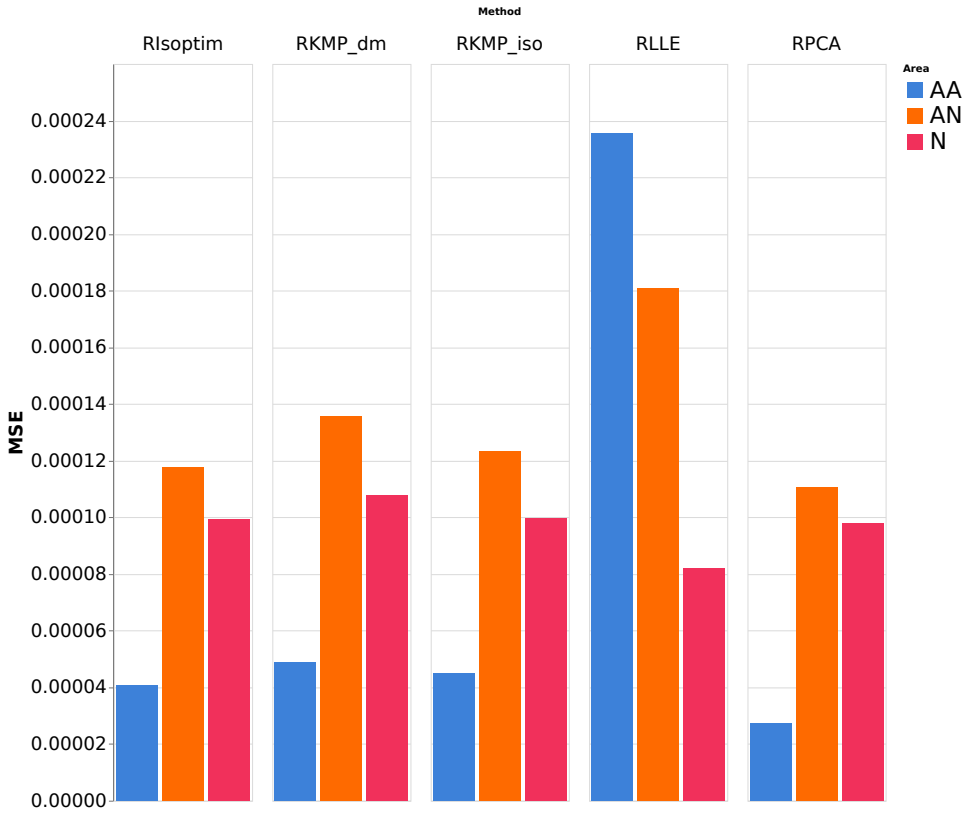

Figure 3.9: Mean-squared errors of robust methods over normal (N) samples, and abnormal ones (split over the normal (AN) and abnormal (AA) areas of such samples) on the large deformation case ($\alpha = 30$).

As the need for robustness to our methods is mostly linked to the percentage of components we altered, results of robust methods and non-robust ones for further experiments are quite similar. Therefore we purposely chose not to delve into it for the rest of our experiments on this dataset.

3.1.4 Fewer Number of Samples and Greater Intrinsic Dimension

Fewer number of samples

We now try to achieve what we did on the reference case with fewer samples ($N_s = 2000$). The low-dimensional manifold (*i.e.* the 3 dimensional half-sphere) will thus be less well sampled, making it harder for all the methods to perform the projection of new samples: MSE will inevitably be worse than in the reference case, and therefore the detection will be coarser. Indeed, multivariate methods rely on neighbourhoods, proximity, or global covariance of the training set to perform their projection/anomaly detection. As the

number of samples decreases, these methods are faced with an increasingly greater challenge.

Figure 3.10 presents the ROC curves of non-robust methods for this test case. As expected, the performance of multivariate methods is strongly affected by the decrease of samples, with FPR values more than halved at a TPR of 10^{-3} for instance. This means that the performance of our test (TPR at a given FPR) is **strongly** linked to the number of samples in our dataset.

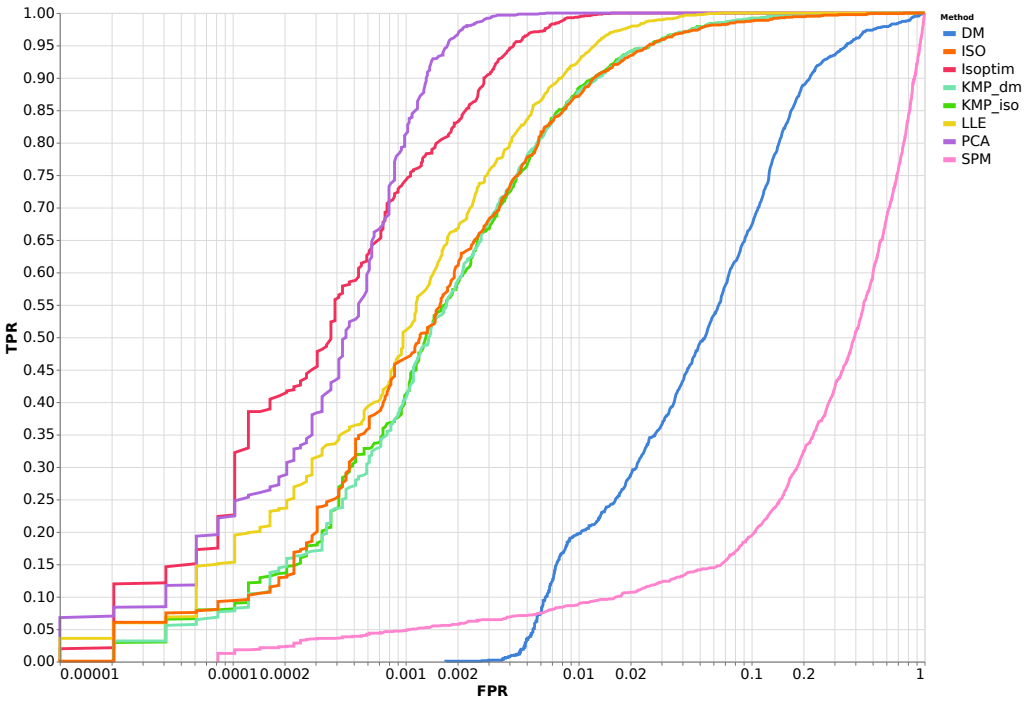

Figure 3.10: ROC curves for all the non-robust methods on the half-sphere dataset, on the fewer number of samples test case ($N_s = 2000$). FPR axis is in log scale.

Greater Intrinsic Dimension

The greater intrinsic dimension test is dedicated to finding the influence of the intrinsic dimension of the manifold on which our data lies. Obviously the intrinsic dimension plays a great role in our datasets and for the difficulty of the task at hand, as it is much harder to correctly sample a manifold of high intrinsic dimension. This experiment is quite similar to the previous one in the sense that it tests the effect of badly sampling our manifold. Here however, this is a much more difficult task, as sampling a 3 dimensional manifold with 2000 points is **widely** better than a 20 dimensional one with 10000 points, due to the curse of dimensionality.

Figure 3.11 presents the ROC curves obtained for our non-robust methods for a test case where intrinsic dimensionality m was set to 20.

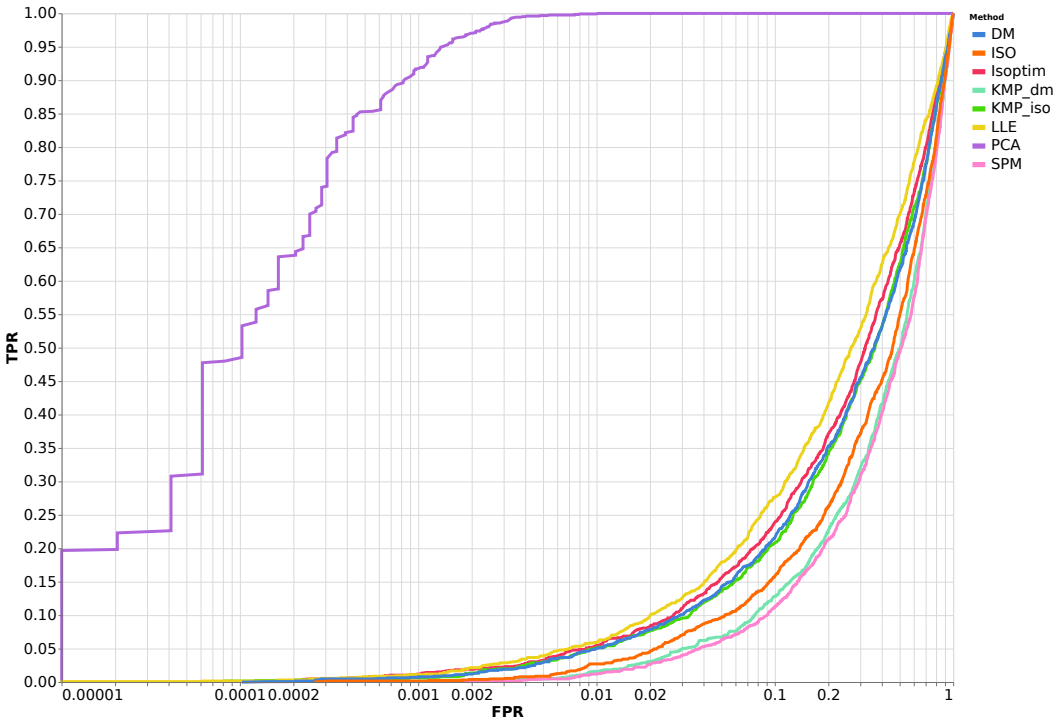

Figure 3.11: ROC curves for all the non-robust methods on the half-sphere dataset, on the greater intrinsic dimension test case ($m = 20$). FPR axis is in log scale.

It is immediately noticeable that every method except PCA has an extremely poor performance with this test case. Non-linear methods might be more affected by the emptiness of the incorrectly sampled latent space, as they mainly rely on the assumption that the number of training samples is consistent with the intrinsic dimension of the manifold on which the data is sampled, and therefore rely on neighbourhoods or *proximity* between samples, which are strongly altered as the 20 dimensional space is mostly empty. PCA is less affected, as its only interest is to find the mapping between the high-dimensional space and the low-dimensional one. This is still a *relatively* easy task for PCA (on this specific dataset) as it is a global model, and as we are still using a linear, orthonormal mapping and PCA does not need for the manifold to be well-sampled to find this mapping.

3.1.5 Large Sample Space Dimension

None of the presented methods were really affected by the increase of the sample space dimension d (even with values similar to real dataset problems, *i.e* $d = 10^5$), which is encouraging as the dimensionality of our samples will tremendously increase as our datasets move toward real dataset. This was more or less expected as we are mainly using methods of dimension reduction

that are designed to cope with very high dimensional problems, and given that we sampled the intrinsic manifold to the same extent as in the reference case. As the results for this test case are extremely similar to the ones from the reference case, we purposely chose not to display them.

3.1.6 Partial Conclusion

With this basic geometric toy example, we have seen a glimpse of what to expect from the performances of the presented methods and the IRLS layer we can put on top. While this dataset is far from being perfect as a non-linear high-dimensional test set, it is full of information on the behaviour of these methods under particular settings. To synthesize, we have learned that:

- PCA is not disturbed by non-linearities in data when **the projection is orthonormal** (non linear projections will be our next focus).
- The IRLS robust layer is often **beneficial**, even with small anomalies, but nearly useless when a method is too well reconstructing the anomalies we are trying to detect.
- High dimensionality of the sample space is a **much smaller problem than the intrinsic dimensionality** of the manifold underlying the data, which can cause any method to collapse due to the vast emptiness of the intrinsic space.
- The sampling of the manifold is the key to a good anomaly detection: **the better the (low-dimensional) manifold is sampled, the better the anomaly detection**.
- KMP methods are most of the time an improvement over their dimension reduction counterparts (*e.g.* KMP_{iso} vs ISO).

Let us now turn to a more realistic dataset on which we will try to confirm our previous findings and go even further into our understanding of the problem we are facing.

3.2 Geometric Images

Contents for this section

The next step in making the synthetic data model closer to real, medical datasets is to deal with images rather than abstract geometrical representations. Therefore, we looked for a way to modelize a manifold of images where we could control the dimensionality as well as the way to introduce anomalies (and thereby the associated ground truths). A quite simple and intuitive way we found to do this is using geometrical shapes images, *e.g.* images of squares or circles, or any given geometrical shape that can be controlled by few parameters from which we can sample to generate a dataset. This is not strictly speaking a direct extension of what we did in the previous section with the spherical dataset, but the idea is essentially the same: obtain a non-linearly distributed dataset (of images) in high dimension that is sampled from few known parameters.

To begin with, we should clarify that all our geometric shapes, unlike mathematical ones have an inherent **thickness**, as any object from discrete geometry. Even more in our setup, as we will use this thickness as one of our parameters. A simple *regular* geometrical shape that requires a few parameters to be described is the trapezium, a quadrilateral with at least two parallel opposite sides. To define a general trapezium, you would be required to provide 4 parameters (for instance the length of three sides and its height), however for the sake of simplicity we only sampled trapezium which are symmetric according to the horizontal axis splitting our images in half (*i.e.* each parallel side of the trapezium is centred according to this axis), leaving us with maximum 3 parameters to sample from (the lengths of the two parallel sides and the height). As we are looking at images rather than pure geometrical shapes we will centre our geometrical forms as to avoid introducing a translation parameter. Obviously, you should also define the angle between one of the side and an axis of the coordinate system. This leads to defining a last parameter, which is a rotation one: each of the trapezium images can be rotated around its centre to form a new trapezium image. Therefore, we are able to sample from a manifold of images whose intrinsic dimension is up to 5.

3.2.1 The dataset

Our dataset is made by sampling the different parameters that rule this dataset (the lengths of the parallel sides, the height, the thickness and the rotation angle) with the chosen number of samples N_s. Once done, we use these parameters (upscaled to match the resolution) to create “high-resolution” (400

by 400) binary images of the corresponding trapezium shapes. To avoid the pitfall of having sparse images (as the trapezium constitutes only a small part of our image), we used a technique called **distance transformation** [88] (precisely Chamfer distance transform) over the binary high resolution image, that defines a distance image in which each voxel intensity is set to the minimum distance between this voxel to a non zero one in the binary image. The high-resolution image is then downsized into a low-resolution (40 by 40) one. The upscaling enables us to have distinct resulting images for closely sampled parameters (*e.g.* a thickness of 2.5 and 2.6 respectively would be indistinguishable in a 40 by 40 image). A random gaussian noise is finally added to the signal, and an anomaly is introduced on abnormal test samples in the same fashion than for the half-sphere dataset 3.1.1, *i.e.* instead of adding noise to the target abnormal area, we add a constant that is a multiple of the noise standard deviation. Abnormal components were (arbitrarily) chosen as a patch in our 40 by 40 image of size 10 by 10 (therefore 1/16-th of the image). In our experiments, the noise standard deviation is set up to be $\sigma = 0.05$ and the amplitude of the anomaly is 4σ.

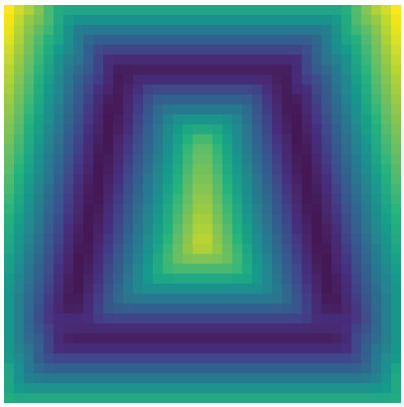

Figure 3.12: An example sample from the trapezium dataset with the Chamfer distance transform (before adding noise).

3.2.2 Trapezium Experiment

Our experiment is designed by sampling 2500 samples of the trapezium dataset using the 5 parameters available. Setting all our dimension reduction methods to match the intrinsic dimension of the dataset, we get the ROC curves of figure 3.13. In this figure, we observe a much greater variance in the multivariate methods results. KMP methods now lead the way, with an exceptionally strong performance of LLE (for which the dataset is now much more adapted). Anomaly detection methods making use of the dimension reduction techniques they are based on have noticeably worse performance than the KMP ones, with the noteworthy case of PCA, which is now one of the weakest method (probably due to the inherent non-linearity of the dataset).

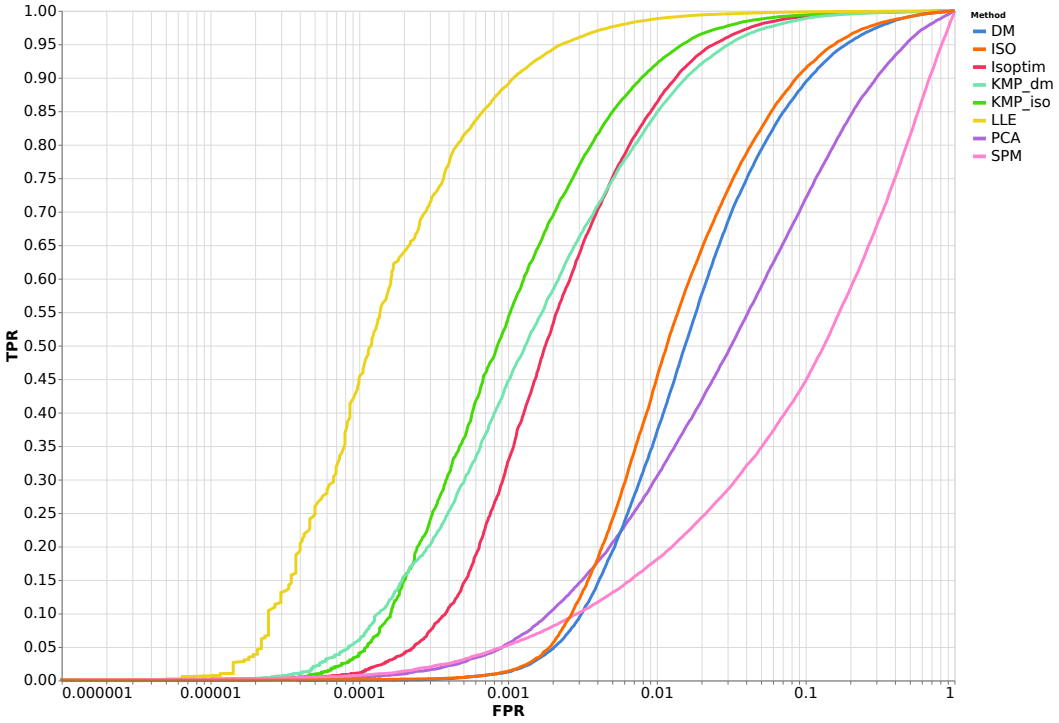

Figure 3.13: ROC curves for all the non-robust methods on the trapezium dataset. FPR axis is in log scale.

While the MSE scores of most multivariate methods were overall quite good, the ones from DM and PCA were rather bad (5 times to 10 times the MSE scores of other methods). The inability of these methods to reconstruct our samples is the prime cause of their failure on this dataset. For DM it is either due to a wrong diffusion parameter (although the same is used for KMP_{dm}), or to the dimension reduction, or to the Nyström extension. Indeed, as is once again obvious on the ROC curves of figure 3.13, the KMP methods corresponding to ISO and DM (KMP_{iso} and KMP_{dm}, respectively) show much better results on the trapezium dataset, suggesting that the embedding part of the ISO and DM methods has a negative impact on our anomaly detection. For ISO, as we increase the dimensionality of our embedding, these distances between embedding points *theoretically* converge towards the one computed by KMP_{iso}. Therefore, the distances computed in embedding spaces are only approximations of the real ones and reducing dimensionality with Isomap is actually not beneficial. For DM as our reconstruction is not based on NWKR but rather on an optimization scheme, the analogy is less simple, but the same phenomenon might be going on too. For both methods, we could also incriminate the Nyström method, as Isoptim's performances are widely superior to the ISO ones on this dataset, while the only difference between the two methods is Nyström's extension. The KMP methods are both using this extension too

suggesting that Nyström's extension might be damaging only when associated with a dimension reduction.

LLE excepted, the robust layer that IRLS provides is here not really helpful as can be seen in figure 3.14, which can be explained both by the small proportion of altered components (around 6%) and the fact that on this dataset, our methods are able to reconstruct the anomaly even if part of it is masked. This might be caused by the fact that some samples in our dataset have a certain resemblance with the anomalies we introduce (*i.e.* their signal is the strongest where the anomaly is located, and close to 0 otherwise). While LLE is constrained to use only neighbours of our sample to reconstruct it, other multivariate are more global in the sense that they will use any sample as long as it is not *too far* from the one we which to reconstruct (which might be the case here). Therefore, without a perfect initialization, the IRLS will ultimately provide a solution quite close to the non-robust equivalent method, or even worse as the noise will be considered more and more abnormal with each iteration (thus introducing false positives) in *some form of overfitting*.

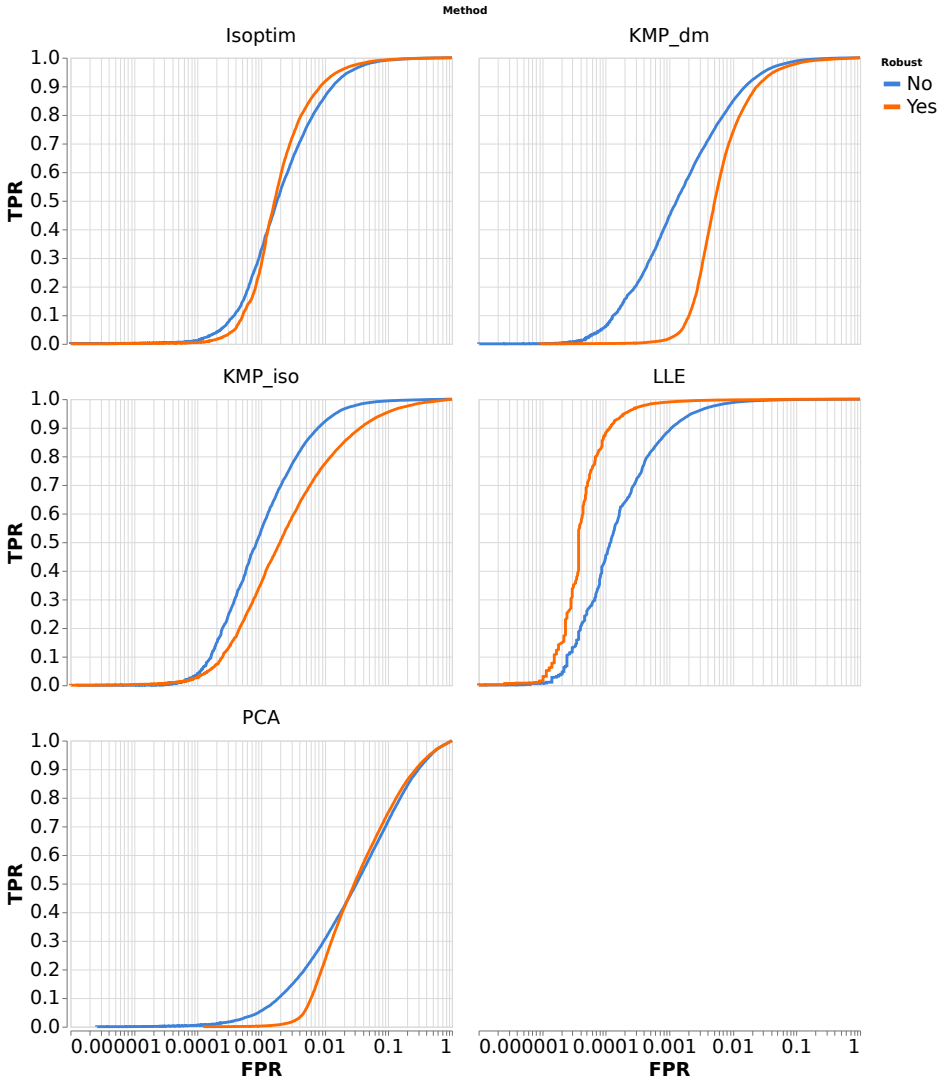

Figure 3.14: Improvement of robust versions vs non robust ones on the trapezium dataset, shown over ROC curves. FPR axis in log scale.

3.2.3 Dimension Analysis

Upon seeing the results over the trapezium dataset, one might object that the PCA performance is probably caused by its inability to model a non-linear dataset with the *correct* amount of dimensions, as it is after all a *linear* method. Therefore a PCA algorithm with more dimensions might perform better than the one using the intrinsic dimension of the manifold. Indeed, intuitively, PCA MSEs (over normal samples) are better and better as we increase the dimension used in the algorithm (as the number of components used to reconstruct our image increases, the reconstruction error can only decrease). We thus designed a test to analyse whether or not a PCA algorithm using more components could perform better than the original one.

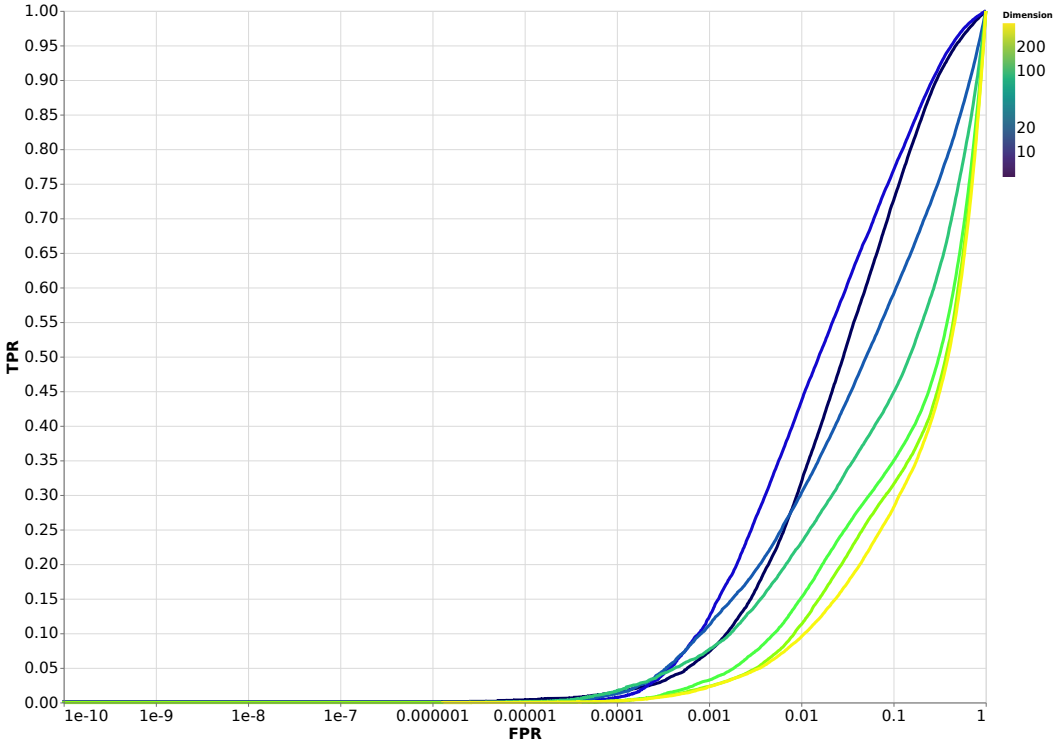

Figure 3.15: Performance of PCA over the trapezium dataset depending on the used dimension shown in ROC curves. FPR axis in log scale.

Figure 3.15 presents the ROC curves obtained by using PCA with 5 to as much of 400 dimensions to reconstruct our data, while figure 3.16 presents similar results for robust PCA. Unfortunately, the results show that increasing the dimension does not lead to better results (although the best result is the one with 10 dimensions, rather than 5). Indeed, as the MSEs presented in figure 3.17 perfectly showcase for PCA, while we improve our reconstruction of normal samples N or normal area of abnormal samples AN, we also better and better reconstruct the anomaly in AA (or conversely reconstruct worse

and worse the original components in AA), therefore making it more and more difficult to detect it.

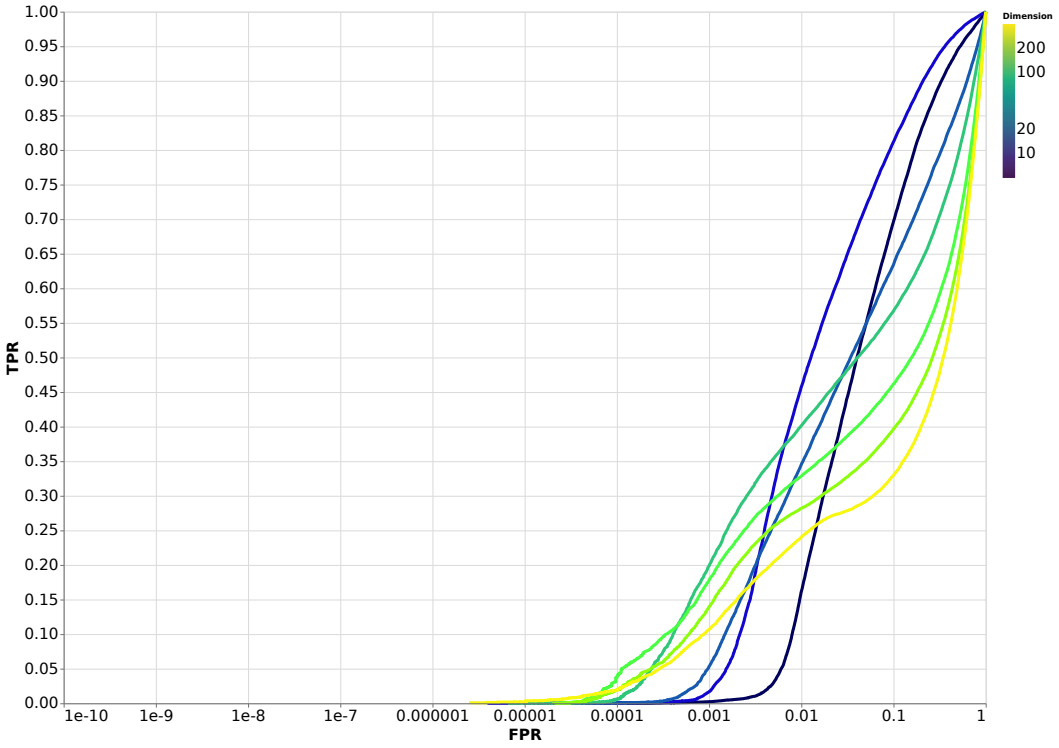

Figure 3.16: Performance of robust PCA over the trapezium dataset depending on the used dimension shown in ROC curves. FPR axis in log scale.

While the IRLS layer do provide *some form* of improvement over the original PCA algorithm (especially in the higher dimensions) as can be seen from figure 3.16, it is mostly skewed by the ability of PCA to reconstruct the anomaly and therefore unable to provide correct results on this dataset. This is close to the previous "overfitting" of non-linear methods (that prevented them to obtain improvement from their own IRLS) and highlighted by the MSE scores of figure 3.17 in which it is clear that with greater dimensions, PCA more and more reconstructs the anomaly.

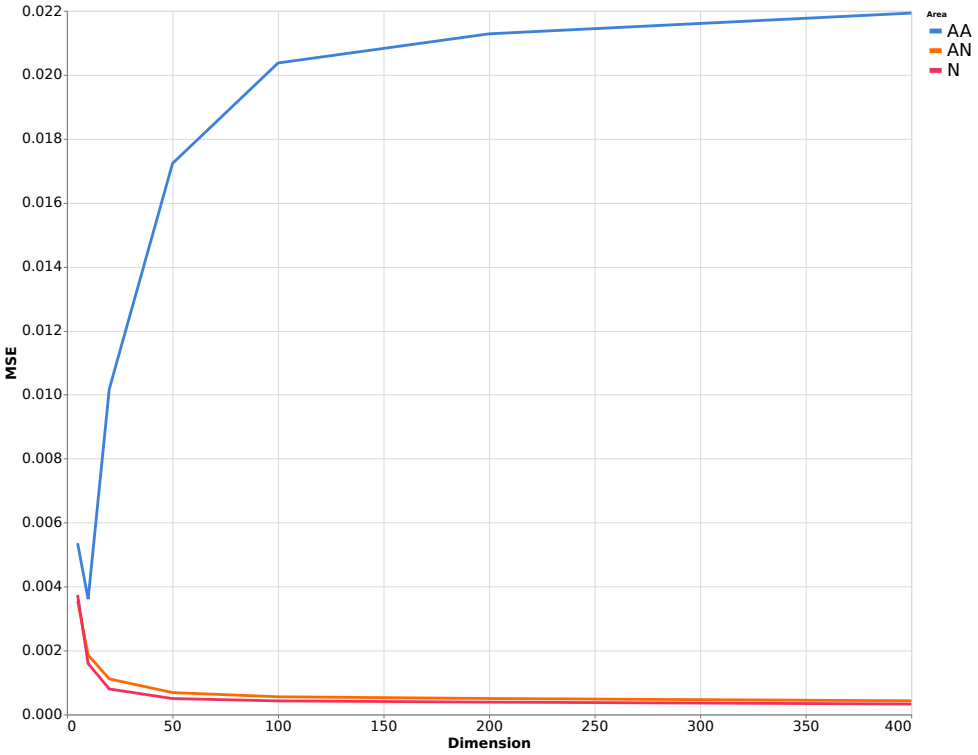

Figure 3.17: Mean-squared errors of PCA over the trapezium dataset depending on the used dimension (x axis). The MSE are computed over normal (N) samples, and abnormal ones (and split over the normal (AN) and abnormal (AA) areas of such samples).

3.2.4 Partial Conclusion

The trapezium dataset has provided us with a more realistic, or at least a more similar dataset to real ones from medical image analysis. The results we displayed from this dataset are both different and consistent with what we experienced with the half-sphere dataset, in the sense that while the best and worst performing methods have widely changed, it happened in a somehow expected way.

- PCA is immensely disturbed by the non-linear distribution of the trapeziums, and is **unable to match the performance** of non-linear methods, even by increasing its dimensionality, as it will only result in overfitting the anomaly.
- Except LLE, **non-linear methods tend to suffer from the same overffiting** that prevents them from obtaining great improvements from the IRLS, due to the peculiarities of this dataset and their non fully local algorithms.
- While LLE was prone to overfitting in the half-sphere dataset, it is not anymore the case on this dataset, thanks to the cut-off provided by the selected number of neighbours.
- KMP equivalents of ISO and DM have stronger performances than their counterparts performing dimension reduction, thus validating the idea that the trouble of actually **reducing dimension is either not required (at best) or penalizing (at worst)**.
- The best performing non-linear method (in its robust version) has been LLE (excepted on a very specific dataset), and will still be further on. Thus we will focus only on its performances in the rest of this work for the sake of clarity.

3.3 Synthesizing the Real World

Contents for this section

In this section, we will take a look at the medical dataset that will be used in our real world test setting. As we mentioned before, we do not have ground truth on the real dataset, therefore we had to make some adjustments (*i.e.* not use the intended test samples as such, but rather *make* our owns) in order to have one and thus obtain quantitative metrics for assessing our results.

3.3.1 MRI Dataset

The Dataset Contents

Our medical dataset consists in around 1500 MRI data extracted from multiple bases. We had a focus on Alzheimer's Dementia (AD) as a research interest, and therefore looked specifically for AD MRI basis as to gather test samples on which to test our methods; however we also looked at other non-specific MRI bases to collect the most possible healthy, control samples to perform our training. We finally settled on four bases:

- The CHUS database is a local basis from a dementia analysis program, with around 200 samples split in healthy controls, diagnosed AD and Lewy's Bodies Disease (LBD), and Mild Cognitive Impairment (MCI) afflicted subjects;
- Alzheimer's Disease Neuroimaging Initiative ADNI [89] is an international database focused on AD but which also provides control samples, from which we extracted around 200 samples (AD and controls).
- The IXI is a database consisting in 600 "normal, healthy subjects" used as control in a psychology study.
- The OASIS dataset provided us with over 500 samples, healthy and AD.

Multiplying bases provides us with an interesting number of samples but also with new issues. These basis tend to have different acquisition protocols, coupled with different set of acquisition machines, which can lead to different contrasts between images coming from different bases. A selection bias is also a risk, as covariables such as patient age or sex are not distributed in the same way across bases. Overall, these four datasets provided us with nearly 1200 healthy samples and around 250 AD. Originally none of these images are sampled from the same space, that is to say size of brains would greatly vary, voxels and structures would not correspond from one image to another.

We therefore had to force every image to belong to the same common space by all registering them together. Registration was performed using an affine registration followed by a deformable one using the diffeomorphic demons [90] algorithm using Advanced Normalization Tools Software (ANTS) [91]. While the affine registration captures all the isometric transformations between our images (translations, rotations, transvections), the role of the diffeomorphic registration is rather to capture local residual deformations that are inherent to each individual. These deformations can bring to the light neurodegenerative pathologies such as AD.

Registration over a common template yields for each image both a registered image than lies in the common space, and a deformation field (that also lies in the common space) from this registered image to the original one. This deformation field is a three-dimensional tensor that indicates for each voxel how the corresponding structure in the registered image must be deformed in 3D to correspond to the voxel of the same structure in the original image. This effectively means that the deformation field associates to each voxel a 3-dimensional vector. As we are not really concerned with such a precision in what we are looking to detect in our test samples, as we do not wish to be sensible to geometric transforms such as rotations and translations, and finally for the sake of computation efficiency during the anomaly detection procedure, we computed the voxel-wise Jacobian of our deformation fields. To be more specific, we computed the determinant of the Jacobian matrices in each voxel.

Physically, this quantity corresponds to the volumetric deformation of each voxel: how much should the volume of the structure underlying this voxel in the registered image be increased/decreased to match the volume of this structure in the original image. This is quite interesting in our case as AD is a disease that strongly affects the brain's structures volumes. To lighten up the writing, as it is widely done in the literature, we will refer to the determinant of the Jacobian matrix itself as the Jacobian, and we will refer to Jacobian images for the Jacobian determinant maps we computed over our samples. As the Jacobian is a *multiplicative* quantity (a Jacobian of two means we need to double the structure's volume) and we wish to *add* and subtract values to our samples, we rather considered the **log-Jacobian** of our deformation fields.

Analysis of Structures Correlations

Before we start creating anomalies to introduce in our control samples, or before we start analysing the real dataset itself, there are interesting points to be made about the multivariate correlations that exist between the volumes of the different brain structures. To this end, we used the brain segmentation provided by the freesurfer [92] segmentation over the template image to split our brain into over 100 anatomical areas. We then computed for each subject the volume in each structure by summing the jacobian values inside this structure. To observe correlations between structures, we computed linear correlation scores across all subjects. These scores can then be shown in the form

of a correlation matrix. For the sake of clarity (as one big correlation matrix would be unreadable), we split the anatomical structures into two groups to be analysed separately: subcortical and cortical structures.

Let us take a look at the correlation matrix obtained over the subcortical structures from figure 3.18. One very apparent multivariate correlation that transpires through this correlation matrix is the strong symmetry of brain structures: left structures are highly correlated to their right counterparts. But symmetry is not the only observable multivariate correlation: some structures have high correlations with other structure than their associated symmetric. While not all structures correlate together, it does appear that there are a lot of spatial correlations in this dataset. It should be noted however that *some* structures don't have strong correlations with any of the other structures, making these structures much harder to reconstruct for multivariate methods using the rest of the brain: in these area, no multivariate model is going to perform much better than a univariate one. To put some perspective to the experiments we will conduct in the following sections, the ventricle area tends to correlate with multiple brain structures while the white matter only does so with a few.

Although it is sparser, the correlation matrix for the cortical areas is quite similar in structure to the subcortical one, indicating that multivariate methods will tend to perform a bit better on subcortical areas than in cortical ones, compared to the univariate model of SPM.

3.3.2 Synthetic Anomalies

To create a synthetic dataset, we had to rely only on the healthy control samples, as if we used real test samples from our collected dataset, we could not be sure of which area of the AD's brain should be considered abnormal due to the disease or not. Actually the problem is already quite hard as control samples display a lot of variabilities, some of which might be attributed to a form of brain affliction either benign or in an early stage. We therefore split our control dataset into a training part (80%) and a testing one (20%). The training part was left as such, while the testing part was split in half to keep the same idea than in the previous experiments: one half of control testing samples, and one half in which we introduced an anomaly (giving us 145 healthy test samples and 145 altered ones). To create an anomaly we could not rely this time on the noise variance as we did not introduced one as opposed to the other synthetic datasets. We can however use the training samples inherent voxel-wise variance as a base for our anomalies, although this will slightly favour SPM (as this model use this exact variance to compute its Z scores). We thus devised two experiments using this variance σ^2 to create anomalies: once in the ventricular area (consisting in 7% of the brain, see figure 3.19) where the sample variance is great and once in the left white matter (around 17% of the brain, see figure 3.20), a more stable region across subjects. In both cases,

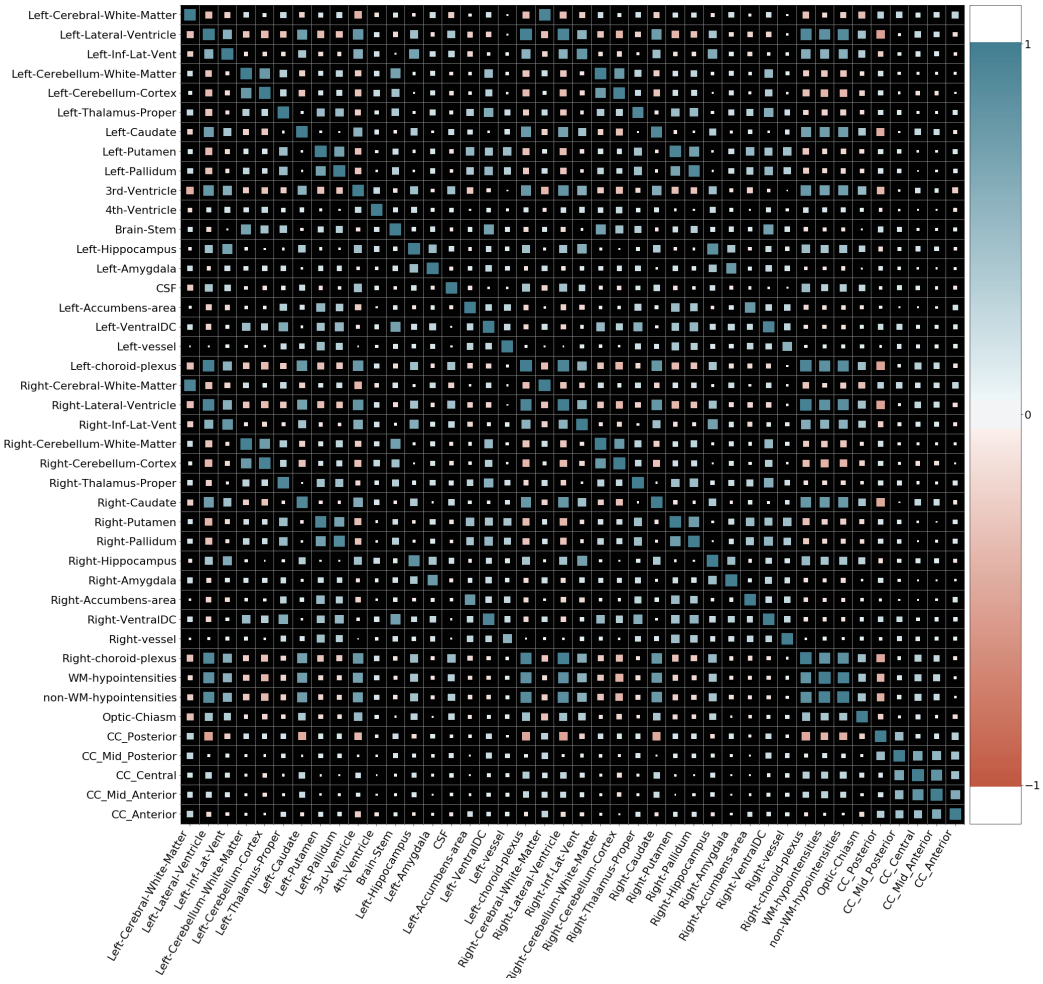

Figure 3.18: Correlation matrix of subcortical structures volumes. A larger square represents a higher correlation in absolute value. A red colour indicates a negative correlation, and a blue colour a positive one.

we voxel-wise added 3σ to each of our test samples. While the value of σ we have chosen to compute *might* favour SPM (as we mentioned), the fact that we add this anomaly to our voxels intensity is also sure to make SPM fail for voxels that constitutes the "left tail" of the distribution (*i.e.* voxels that would otherwise have a negative Z score with SPM if no anomaly was introduced).

3.3.3 Results

Ventricular Area

As previously stated, our first experiment focused on the ventricular area. Figure 3.21 presents the ROC curves obtained on this test for SPM, RLLE, RPCA and the PCA-based method [1, 35] we mentioned in section 1.3.1 denoted as

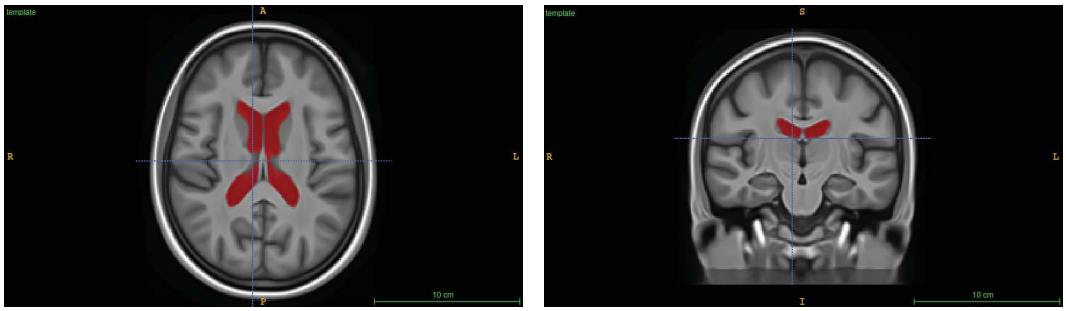

Figure 3.19: MRI template constructed as a "mean" of our registered images. The red area corresponds to the ventricles.

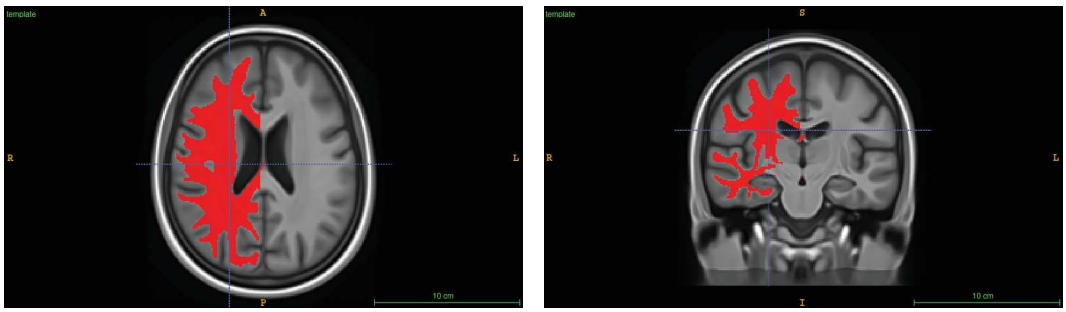

Figure 3.20: MRI template constructed as a "mean" of our registered images. The red area corresponds to the left white matter.

RMIX. All PCA based methods were set to 50 components, while RLLE was set to 20 neighbours.

Note on RMIX: To recall, this method goes a step further than the robust PCA algorithm by modelling the linear subspace obtained by PCA on the training set in order to correct projections for abnormal test samples by making sure their low-dimensional components stay consistent with the training set distribution embedding. We found this method to provide a useful improvement over RPCA on this dataset (while the difference was negligible on the previous synthetic datasets), and therefore added it to the list of best performing methods.

On this experiment the result provided by RPCA is widely incorrect, as it suffers from the same *overfitting* as previously: by using widely different (from the training set distribution) values for its coefficients used in the linear combination of eigenvectors, it is able to reconstruct the anomaly that as been introduced in the ventricles. RMIX on the other hand, performs a correct estimation of the training set coefficients distribution, and therefore does not able to reconstruct the anomaly as well as RPCA. RLLE has a similarly strong (or even stronger) result on this test case. Both RMIX and RLLE largely outperform SPM, even though SPM performance is consistent with the amplitude of the anomaly we introduced. This indicates that the multivariate correlations

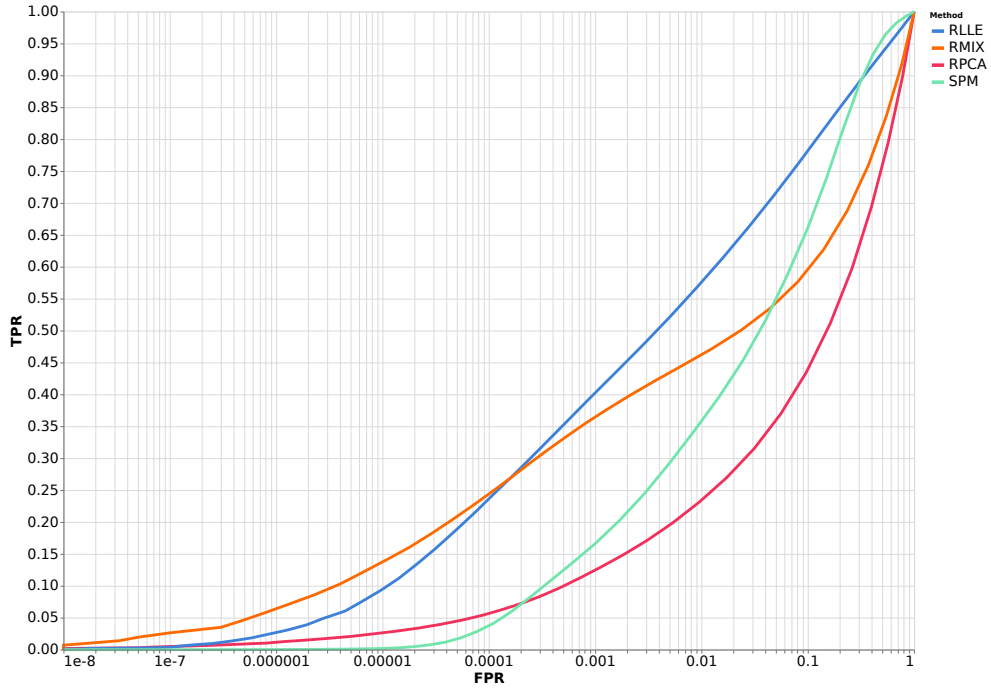

Figure 3.21: ROC curves for the best performing methods on the synthetic dataset created over healthy samples, with an anomaly introduced in the ventricle.

between the ventricle area and other parts of the brain have been correctly captured by those multivariate methods. This can be linked to the observations of the previous section, in which we stated that the ventricle areas had strong correlations with multiple brain structures, making it easier for multivariate methods to reconstruct an abnormal ventricle based on the rest of the brain structures. Still, it is noteworthy that the performance of SPM is highly better than in our previous synthetic datasets, which is easily explained by the specificities of this dataset: while in other synthetic datasets the MSEs of SPM were 4 to 5 times larger than our worst multivariate method, here the MSEs of SPM are only 2 times larger at max. This, added to the fact that (as we have seen on correlation matrices) many brain structures do not present strong correlations with other brain structures (outside their symmetric peer), makes this dataset a not so *non linear* one, or a very packed one, much more suited for the univariate model of SPM than the previous ones.

White Matter Area

In this experiment, we altered the left part of the brain white matter. This is a much larger area than the ventricles but, as mentioned before, also more stable. Results on this area are also much more close, as showcases figure 3.22: while

the performance of our methods are consistent with the previous experiment (with RMIX and RLLE being the top ones, and RPCA and SPM behind), the difference between the best and worst performing methods is now much less pronounced.

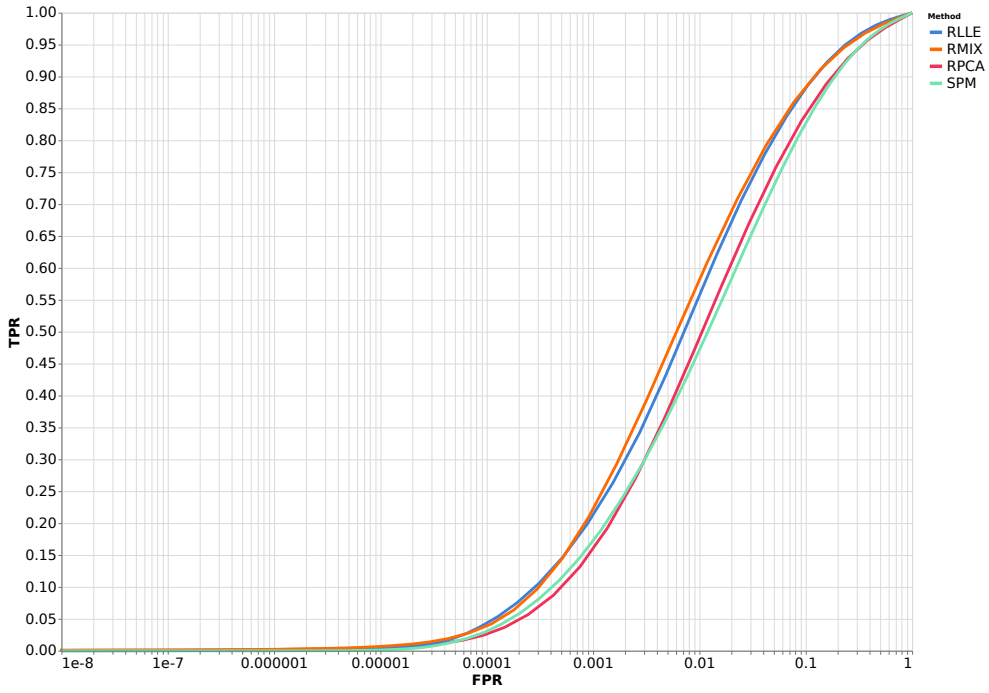

Figure 3.22: ROC curves for the best performing methods on the synthetic dataset created over healthy samples, with an anomaly introduced in the left white matter.

This is a great illustration of how widely the localisation of the abnormal part is important for our detection results: some areas will be much easier to detect than others, even with the same order of magnitude for the anomalies we introduce. Just as for the ventricle result and related to the previous section, this effect can be caused by the amount of correlation between the area we are investigating and the rest of the brain (as an area completely uncorrelated could have an even worse anomaly detection performance with a multivariate method than with an univariate one).

Subsampling Experiment

Just as for geometric datasets, we can look at the influence of the number of samples in this dataset. To this end, we greatly reduced the amount of healthy samples available to perform the training step of each methods, from over 1200 to just 200 (with a similar reduction for the validation set), by random selection. To control for the variance in our data (*i.e.* the different

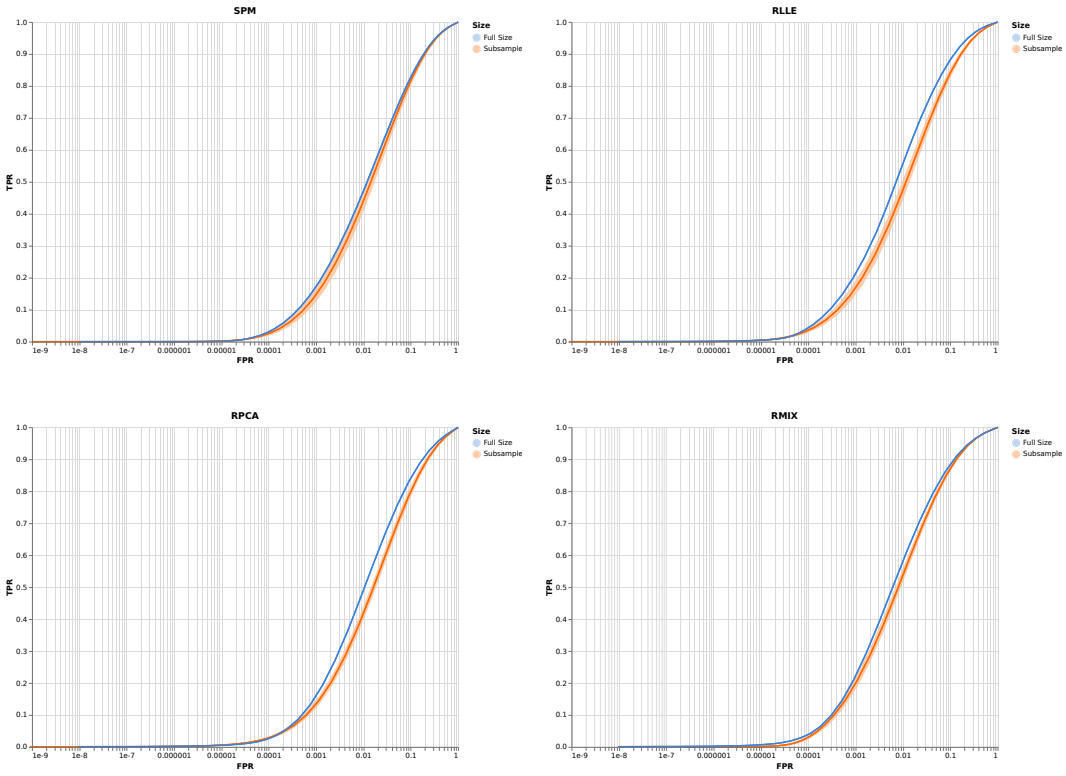

Figure 3.23: ROC curves for the subsampled training compared to fully the sampled one over the white matter experiment. Top left: SPM, top right: RLLE, bottom left: RPCA, bottom right: RMIX

result we would obtain by selecting another 200 samples rather than these), we repeated this experiment 20 times in order to compute mean ROC curves, with 95% confidence intervals. Figure 3.23 showcases the effect of subsampling the real dataset in the white matter area experiment.

It is quite apparent that training on the full set of healthy samples improves results. But not all methods show the same improvements: multivariate methods tend to be more impacted by the fewer number of samples, which is intuitive: SPM will not profit much from more samples in its estimation of the sample mean and the ensuing residual variance, but more complex, multivariate models are able to capture more of the data correlations and neighbourhoods methods will provide closer neighbours. Now, this is obviously a drastic experiment, in which the number of samples has been reduced by a factor of 6, and we would not have seen significant gains by "only" using 1000 samples for training. But it does suggest that with a similar increase in factor of the number of samples, multivariate methods could see another improvement, *provided* that we are not already at the plateau of the learning curve for these methods.

3.4 Conclusion

The whole process of creating synthetic datasets on which to test our methods and to analyse their results has been, by far, the most challenging part of this work. To create datasets that are interesting for us to test (*i.e.* non linearly distributed, non-trivial, over which we can introduce an anomaly easily) has provided us with as much insights on the way our methods work and what to expect on further datasets than the results we obtained over these datasets themselves. By gradually increasing the complexity of the model underlying the synthetic data we generated, we have been able to find results consistent or comparable from one test to another over more and more realistic datasets until we reached one that is directly based on the real medical dataset that will be the object of our study in the following chapter. Here is a brief summary of what we learned over the course of these multiple experiments:

- Non linear methods in the sense of what we presented earlier do have a strong interest over non linearly distributed datasets.
- More important than the distribution in the low-dimensional space, the embedding function between the ambient space and the embedding is crucial for the performance of linear methods.
- A phenomenon resembling what we could call "overfitting" is afflicting most of our methods, and breaking the idea that "the better the reconstruction, the better the detection".
- Non linear methods that do not rely on dimension reduction to perform anomaly detection tend to have better results than otherwise, questioning the interest of Nyström method associated with reducing dimension.
- The IRLS layer that provides robustness is almost always beneficial for all of our methods.
- SPM will tend to work much better on real world datasets.
- The area in which anomalies are located has a great influence over the confidence with which we will detect it.
- Increasing the number of samples could benefit multivariate methods, while SPM plateaus much sooner.

Let us now introduce our work on the real medical dataset, which is designed to test our methods in a volumetric analysis of dementia afflicted patients setup.

Chapter 4

Medical Dataset

This Chapter contains:

This chapter is dedicated to the study of the real MRI dataset we introduced in section 3.3. To recall, this dataset contains around 1500 healthy control MRI and nearly 250 Alzheimer's Disease (AD) afflicted MRI. As we already mentioned, our greatest concern while dealing with this dataset is that we lack a ground truth to know whether the detections that we will make using our methods are relevant or not. In fact, if a ground truth existed for those data, it should be specific to each subject. Indeed, the dementia we are studying can take several forms to affect the brain. Moreover, the magnitude with which the brain will be affected will obviously be linked to the time since the disease has been afflicting our subject. Finally, each subject can be afflicted with other brain diseases in addition to being affected by AD. Only a complete medical history of each patient could allow to distinguish between anomalies caused by AD and ones due to other diseases. An individual ground truth would be thus *extremely* cumbersome to provide, and nearly impossible to make at a voxel level. Therefore, we made the choice of detecting anomalies on individual subjects afflicted with AD, but to present results averaged over groups of subjects. Biomarkers for AD are relatively well known now (hippocampus, amygdala, entorhinal cortex, temporal lobe, etc.), and we expect our methods to detect them in subjects as well as areas not specific to the disease. This way, by averaging over the population, mostly the biomarkers should appear, as anomalies not related to AD differ from subject to subject.

4.1 Coupled Anomaly Detection

Contents for this section

Before we showcase any of our results, we must mention that although the diffeomorphic registration provided by ANTS has been *globally* quite efficient and nearly excellent on large areas, some images showed examples of failed registration for small structures, especially for dementia afflicted patients. These structures showing large atrophies were not correctly registered (and thus not detected) although they can be of crucial importance for the kind of neurodegenerative disease we are interested in. Therefore, we performed our task of anomaly detection not only on the log-jacobian images, but also directly on the corresponding registered images: registered images of healthy samples become our training set, over which we project the registered images of test samples. As registered images or neither normalized inherently nor by the registration algorithm, we performed standardization (removing the mean and dividing by the standard deviation for each sample) of all registered images before the anomaly detection process.

Typically, if the registration had gone perfectly for each of our images, then registered images would be very much alike. However, in the instances where the registration has failed in some structures, then it will most likely show over the registered image, that will not look like the other ones. As this problem is much more likely to happen with the abnormal subject (as they are the ones deviating from the much larger healthy group), this provides us with another way of detecting anomalies, complementary to the ones over log-jacobians in area where the registration has failed. To perform our coupled anomaly detection, we will effectively perform each one individually beforehand, and then compute a "coupled" Z-score as a logical OR, which is to say that we will compute the maximum absolute value between the two Z-scores and return the corresponding Z-score. This way, when we will threshold our score to obtain binary detections rather than quantitative Z-scores, we will make our decision based on the one of higher magnitude. This is done with the idea that if we did not detect anything in the log-jacobian but detected something in the registration it is probably because the registration failed, and therefore we need to detect it.

4.2 Alzheimer's Dementia

4.2.1 Group Detections

As we indicated earlier on, we will start by presenting results over the whole dataset. This means that for each subject (and each method), we will perform our anomaly detection by thresholding its Z-score map at a given value common to every subject (and every method). As we are not able (due to our coupled detection process) to detect whether an anomaly is coming from an atrophy or an hypertrophy of the corresponding area, we will perform this tresholding on the absolute value of our Z-score, providing us with unsigned detections. As single subject detections are very conservative at the classical threshold of 3, and as we are here in a group context, we will use a more permissive value of 2 for our threshold. This might introduce more false positives in our detections, but those false positive should be spatially uncorrelated, and should fade out once averaged over all our samples.

Figures 4.1 to 4.3 present the **mean detection rate** (MDR) over the AD dataset for the top three methods: SPM, RMIX and RLLE. The intensity of each voxel denotes the average number of time this voxel has been detected across every tested AD subject. For an anatomical reference, this MDR is superimposed to the template we computed with our registration process: the mean of our registered images.

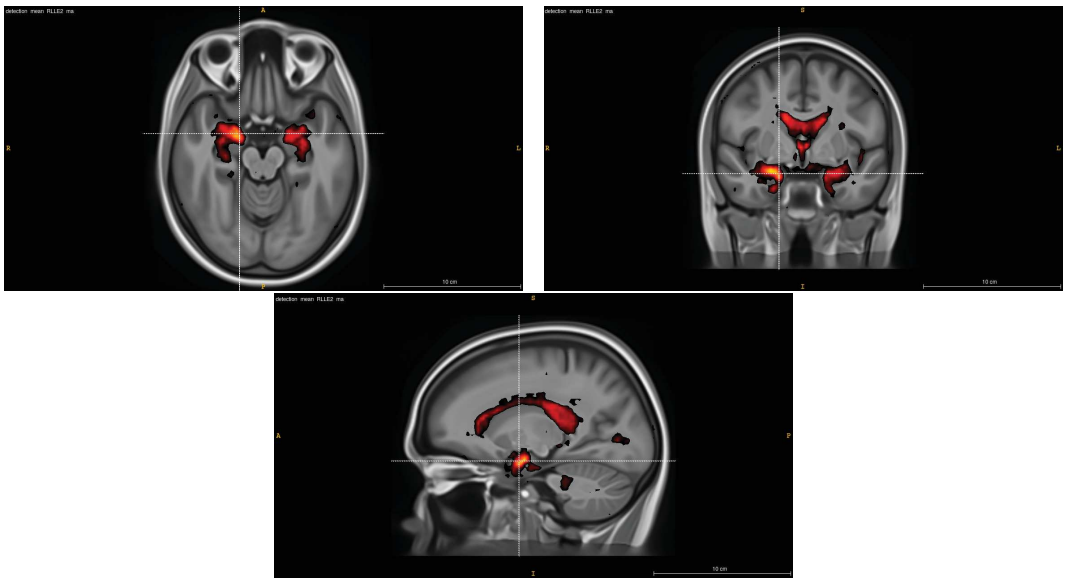

Figure 4.1: Mean Detection Rate of **RLLE** overlayed over the template image. MDR is thresholded to appear only for values superior to 15%, and to be white over 50%.

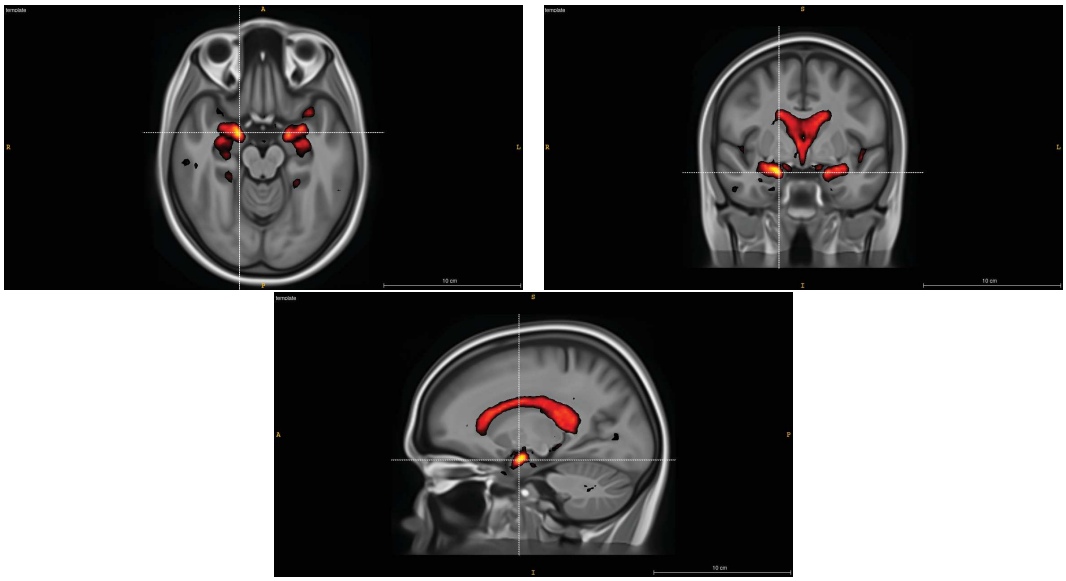

Figure 4.2: Mean Detection Rate of **SPM** overlayed over the template image. MDR is thresholded to appear only for values superior to 15%, and to be white over 50%.

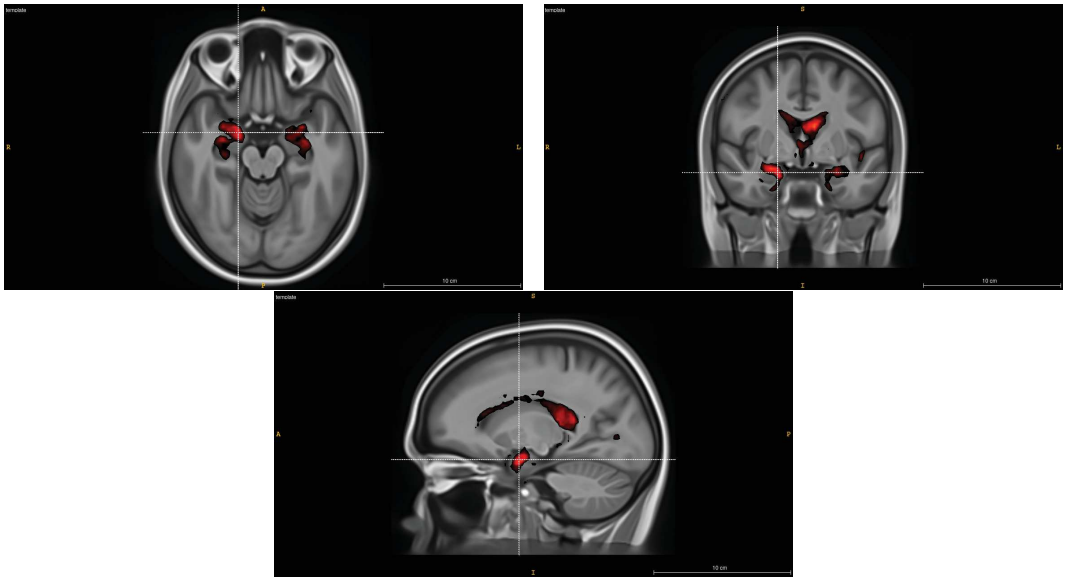

Figure 4.3: Mean Detection Rate of **RMIX** overlayed over the template image. MDR is thresholded to appear only for values superior to 15%, and to be white over 50%.

What is quite apparent on these figures is that, although the detection rates may vary between the three methods, they do seem to focus (for the most part) on the same areas, *i.e.* all the methods tend to agree on what has been deemed abnormal in our test subjects. Furthermore if we take a closer

look to the areas that have been commonly detected, we get the entorhinal cortex, the hippocampus, the Amygdala and the insular areas, that are all classic markers of AD (in the chronological order of evolution of the disease). We also detect the ventricles with various degrees of MDR. Although it is not a specific marker to AD, it is a marker of dementia in general (and elderly age). Some additional detections are made in the white matter (near the precuneus or the fusiform), last structure to be afflicted by the dementia (after having afflicted the grey matter structures, the disease attacks the communication way between these structures). Globally these detection maps are satisfying: they prove that the three methods perform *relatively* well on this dataset, and are able to identify biomarkers of the disease as abnormal components of the brain. In more details, SPM and RLLE presents quite a bit more detections than RMIX (that peaks at a MDR of 35% in the amygdala, while SPM and RLLE peak at nearly 50%). RMIX pattern of detection is globally much more diffuse. Of course, without a ground truth, we can only speculate over the number of false positives that have been detected by each method, and whether or not RMIX is making less of them than SPM or RLLE, but at this threshold, detections for RLLE and SPM seem to be much more concentrated over regions of interest than for RMIX.

If we now take a closer look at the detections of SPM and RLLE, we notice that although their performances are very similar near some of the biomarkers we mentioned (hippocampus, amygdala and insular area), three key differences are noteworthy:

- First in the enthorinal cortex, which is suspected to be the first area afflicted by AD. The MDR of RLLE is arround 1.5 to double the MDR of SPM in this structure, SPM nearly ignoring this interesting structure.
- In the ventricle area then, that as we mentioned is not a particular biomarker of AD, the opposite happens: SPM now detects in around 1.5 to double of the subjects that ventricle voxels are abnormal, some of which might be false positives as AD is not supposed to have this much effect on the ventricle area.
- SPM has very few detections in the white matter (near interesting areas of the temporal lobe for instance), which is nevertheless known to be afflicted by AD at some stage of the disease.

These observations are quite satisfying for both RLLE and SPM, with a bit of a disappointing result for RMIX. RLLE has *some arguments* over SPM, but is not completely ahead as a top contender.

4.2.2 Clustering Detections

One interesting way of further analysing the detections we made over our test subjects is to try to cluster them in smaller groups that maximise the

closeness of Z score maps. Indeed, by doing so we will aggregate together all samples that have similar patterns of neurodegeneration, therefore hopefully constituting quite different groups with even more contrasted MDR maps (very high MDR in some structures and very low elsewhere). This idea is motivated by the suspected medical hypothesis of there being not just one form of AD, but at least three different kinds, affecting differently the brains with some common patterns.

Therefore, for each of our three methods, we used agglomerative clustering (a form of hierarchical clustering that minimizes a linkage distance inside and between the obtained clusters) to provide for each of them three clusters based on similarities between Z score maps. We then computed the cluster-wise MDRs in the same fashion than for the global AD dataset to test the previous hypothesis. Obviously these clusters have no cause to be similar across our three methods, nor to be balanced in sample size. Thus we chose to order them by decreasing size for each methods (we will specify their respective sizes in our analysis.)

To present results for three methods, over three clusters on a 3D images dataset would involve the analysis of at least 9 slices of MDR overlayed to the template image, which would be very cumbersome. We instead chose to visually inspect our MDR scores in terms of areas afflicted as to synthesize the informations inside each of our cluster MDR map, and to present them in table 4.1. This table reads as such: for each method (vertically), clusters are represented in cells (horizontally) in which the structures showing up in MDR maps are listed with the MDR ranges found for these structures.

Method	Cluster 1	Cluster 2	Cluster 3
SPM	• Ventricles (0.4-0.5) • Amygdala (0.4-0.5) • Parahippocampal region (0.3-0.4) • Insula (0.2-0.3)	• Ventricles (0.7-0.8) • Amygdala (0.4-0.5) • Parahippocampal region (0.4-0.5)	• Amygdala (0.3-0.4)
Size	• 114	• 79	• 55
RLLE	• Ventricles (0.5-0.7) • Amygdala (0.3-0.4) • Entorhinal Cortex (0.3) • Hippocampus (0.3) • White matter (0.3)	• Hippocampus (0.4,0.5) • Amygdala (0.4-0.5) • Anterior insula (0.3-0.4)	• Amygdala (0.4-0.5) • Hippocampus (0.4,0.5) • Entorhinal Cortex (0.3)
Size	• 147	• 66	• 35
RMIX	• Ventricles (0.4-0.6) • Amygdala (0.4-0.5) • Hippocampus (0.4)	• Hippocampus (0.3,0.4) • Entorhinal Cortex (0.3)	• Amygdala (0.3-0.4)
Size	• 133	• 63	• 52

Table 4.1: Clusters obtained for each methods on the AD dataset with most afflicted structures and associated MDR listed.

By analysing the results from table 4.1, we notice immediately that even though clusters do not fully match between methods, there is a clear intersection between them that is confirmed numerically (up to 90% of correspondence between the clusters of RLLE and RMIX, and between 50% and 70% between SPM and RLLE). On the sample size matter, we also notice that SPM clustering is much more uniform than the ones from RLLE and RMIX. Let us now look into each method individually:

SPM

The first cluster for SPM is a bit disturbing at first sight, with most detections made in ventricles and the amygdala. While the affliction of the amygdala is consistent with AD, the ventricles detections are probably not all related to the disease. Although we detect parahippocampal areas, few detections are made in the hippocampus or its *direct* vicinity, although it is the most biomarker for AD. No detections either are made in the entorhinal cortex, which is one the first structures affected by AD. All of this would suggest a rather bad detection of these individuals by SPM, coupled with subjects that have been afflicted by AD for a long time. Some detections have also been made around the whole insula, which usually rather is a biomarker of Dementia with Lewy Bodies (and therefore a bit surprising here).

The second cluster is mostly constituted of detections in the hippocampus or parahippocampal areas, amygdalae and the anterior insula, which is very consistent with AD in an hippocampal form.

The last cluster only presents detections for around a third of the subjects in the amygdala, which is not consistent with any form of AD as at least the hippocampus or the entorhinal cortex should also be afflicted.

RLLE

The first cluster for RLLE is consistent with subjects that have suffering from AD for a long time: all of the structures near the temporal lobe have been afflicted (entorhinal cortex, hippocampus, amygdala), with a progression of the disease even in the white matter near the temporal lobe, the precuneus and the fusiform, where detections have also been made. All these biomarkers are consistent with a long- term set AD.

The second cluster is restricted to the hippocampus, amygdala and anterior insula, again strongly consistent with an hippocampal form of AD, centred around the medial temporal lobe.

The last cluster is consistent with early and near-prodromal forms of AD, with detections in the entorhinal cortex, the amygdala and the hippocampus, representing the chronological progression of the disease.

RMIX

Unfortunately, even by clustering, MDRs are still quite low for RMIX. The first cluster is focused on amygdalas and hippocampi, that *could* represent an already developed stage of AD. The second one is closer to an early stage of the disease with detection of the entorhinal cortex along the hippocampus. Lastly, as for SPM, the last cluster only detects amygdalas, which is not quite consistent with any form of AD.

4.3 Conclusion

The real dataset has shown results widely different from what we saw on synthetic datasets, although the experiments we conducted over these datasets have helped us get a better understanding of what is happening now.

- **SPM has a good performance on real data** (which is no surprise given how frequently it is used for medical purposes), but is very focus shifted on the ventricles, that are not of peculiar interest for us.
- No method is detecting consistent anomalies in a given area, although we are dealing with a dataset of abnormal subjects only. This suggests that **being afflicted by AD does not always imply strongly abnormal brain volumetry**.
- **RMIX has quite an underwhelming performance** on the dataset, with very few detections. This suggests that anomalies are distributed in a different way than in our synthetic tests which makes the linear projection harder.
- The **differences between SPM and RLLE, albeit small in global**, are somehow **located on areas that are biomarkers of the disease** (where RLLE was a clear winner), and therefore of strong interest in our analysis.
- Our clustering experiment was not really successful for SPM and RMIX, probably because it was skewed by the low detection rates of these methods for the regions of interest. For RLLE, however, it seemed to show a **chronology of the disease progression**, that even if it was not the primary focus, was very interesting.

Chapter 5

Conclusion

5.1 Conclusion and Future Work

5.1.1 General Conclusion

Overview

The work presented in this document is the direct continuation of several other theses conducted in the research group that hosted me. Although we mainly focused on the excellent work of Torbjørn Vik [1], we had to mention all of them for their contributions, without which we would not have had points of comparison nor the source of inspiration from which many of our methods stemmed.

This thesis had several objectives when it started, some of which we had to give up or adapt a bit throughout the years to account for the difficulties we had to face. It originally had a strong focus on medical images application, and while we sometimes had to divert our attention from the medical perspectives to further investigate the methods we created and modified, we always kept in mind that they should have to serve the purpose of helping doctors in their medical image analysis. To this end, our main goal was to devise non-linear models in order to perform anomaly detection *via* statistical analysis, over medical image datasets. Furthermore, this anomaly detection was to be a voxel-wise, multivariate one, that should have the strongest possible performance at low specificity scores.

With such specifications in mind, our work was successful: we actually supplied several methods with these characteristics, that presented good enough results on synthetic datasets to be tested over a real one. Although some of the presented methods lacked originality as they already had been developed in different ways and with other problematics, we did provide new, original ones as fresh contributions to the field that we linked to the previous ones in a framework that had not been considered before. Moreover as another contribution, we transposed the robust layer that had been well established and documented on linear methods to non-linear ones, in an unprecedented move.

Contributions

Our main contribution was to fuse together the paradigm of multivariate projection and statistical testing with the main hypothesis that we made during this thesis and that is globally accepted in the image community, *i.e.* the manifold hypothesis. As this hypothesis underlines the idea that images lie upon non-linear manifolds, we introduced non-linear algorithms to perform our projection.

Non-linear algorithms however, do not have the nice properties of linear ones for out-of-sample extension or reconstruction problems. Therefore we had to provide extensions and reconstructions using the most accepted tools from the literature. By this mean, we created new methods for manifold pro-

jection. Using Isomap or the Diffusion Maps in such fashion had not really been done before, although some experiments had been done with close methods. Borrowing the statistical testing used with linear methods of projection to perform an anomaly detection, we completed the first algorithms that fulfilled the main objective of our thesis.

In an effort to improve upon this early non-linear methods of anomaly detection, we went a step further in our manifold projection quest by cutting out the dimension reduction step. To do so, we designed a kernel-based method of projection that relied on the non-linear kernels that are computed with our dimension reduction methods. The projection was then performed as a linear combination of training samples, with weights computed in a non-linear fashion using the different kernels. This was the second step in our work, that somehow united both paradigms and all kernel-based methods of dimension reduction in a new framework.

Finally, as we had noticed all the improvements brought to linear methods by the robust layer the IRLS algorithm provides, we managed to adapt it to our non-linear methods, as to be able to deal with the same widely abnormal images that linear methods can deal with, and see the same gains. This completed our efforts to supply non-linear methods of anomaly detection on images, providing us with multiple robust algorithms that are more suited to non-linearly distributed datasets.

The last part of the thesis was dedicated to providing numeric results that could bring informations on each of our methods performances and how well they fare compared to linear ones. Indeed, the real medical dataset we had a focus on since the beginning of the thesis lacked a ground truth that would bring this to light. We thus designed several well controlled synthetic datasets in a first time, that provided plenty informations about our methods and how that would transpose to the real dataset, whose analysis we developed in a second time. The final results were investigated with the help of doctors, and while not reaching a definitive conclusion, they were largely optimistic.

5.1.2 Insights Gained Through This Work

We now take a look back to what we learned throughout this thesis. We knew from the start and section 1.2 that linear methods such as PCA and SPM would be disturbed by non-linear datasets and that we could strongly benefit from using multivariate methods upon spatially correlated datasets. But we also learned in this section that although non-linear methods of dimension reduction are the most efficient for retrieving the correct geometry of the underlying manifold, PCA will be able to provide a good dimension reduction given enough intrinsic dimensions. This finding somehow reopened the debate on linear methods of dimension reduction being as efficient as non-linear ones for non-linearly distributed data. In the rest of chapter 1, we learned that it would be more difficult for non-linear dimension reduction algorithms to

be “completed” into manifold projection ones, or rather less elegant than the inherent ones from PCA and SPM, reinforcing this sentiment.

However, from chapter 2 we did manage to assemble several of those non-linear manifold projection methods, and started to notice some similarities (the space of optimization, or the form of the reconstruction for instance) between them, paving the way for a more elegant framework unifying the non-linear methods. This resulted in a split of non-linear methods into two categories: ones using dimension reduction and others which do not. The latter ones proved to be much more easily transformed into robust ones with the help of IRLS.

We learned a lot from chapter 3, which was made for this purpose. The multiple synthetic datasets and the different parameters we could change to test our methods allowed us to gather crucial informations about the comportment of all methods. At first, we learned that PCA was definitely able to modelize non-linear datasets depending on the projection between the inherent manifold and the ambient space. Moreover, the dimension of the inherent manifold was confirmed to be the dimension of the space we need to correctly sample (rather than the ambient one, which was also confirmed to be of very little importance). The idea that to perform a better anomaly detection one should provide a better reconstruction was quickly debunked by our MSE analysis. Thereby robust versions of non-linear methods (which provide even worse reconstructions) presented very significant improvements, much alike the RPCA one. We also learned that the dimension reduction step was either not required or even penalizing, which suggested that for our purposes, it was better to not perform it, even though the obtained embedding was of great quality. All of these results were asserted with the real world-like synthetic dataset, with a kernel manifold projection method being (again) the strongest contender for non-linear algorithms. The most important lesson from this chapter therefore is that as we do not use the dimension reduction embedding, we also do not need to compute it. LLE being the strongest performing method is also a good indication that the datasets we tested (including the real one) are sampled *correctly enough* so that for our test subjects we can find a set of *good enough* neighbours. However the distribution of these samples is not uniform enough for methods with a more complex modelization (such as the KMP versions of ISO or DM) to perform better than LLE.

The last chapter 4, where we looked at results on the real dataset corroborated with these findings, with RLLE being more coherent in group detections than SPM or any PCA based methods. This suggested that we did have non-linearities in our dataset, and that multivariate methods were better able to capture the spatial correlations than univariate ones. Globally, all that we have learned with those methods and datasets makes a good case in favour of multivariate, non-linear methods for adequate datasets, provided that they are correctly sampled: the better the sampling, the more complex the model we can use.

5.1.3 Perspectives

Short-Term Perspectives

One of the focuses of the thesis was the ability to incorporate covariables to our non-linear models as it can already be done with some of the linear ones. Although we tried very immediate approaches to perform this task, such as including covariables in our distances computations, or introducing an intermediate layer on our dimension reduction corresponding to some of them, these ideas did not improve our algorithms results. Obviously that is not to say that nothing can be done in order to provide a useful incorporation of covariables in our model. One short-term perspective could be to perform a more elegant, in-depth modelization of the covariables effects on our data (which is known to be non-linear), as to perform a regression of our dataset corresponding to this model, therefore eliminating the influence of said covariables.

As we have very different results for several methods, that all bring different types of information, we could issue another one that gathers most of the interesting features of these methods, and few of the drawbacks by using *ensemble* techniques, which group all results together in some sort of majority voting to deliver a *supposedly* better performing one.

A last short-term perspective would be to reduce a bit the difficulty of the task we are faced with. Doctors are as interested in what structures of the brain is afflicted as having a complete voxel-wise anomaly detection. Therefore, based on the results of our projection, we could provide a structure-wise anomaly detection (indicating for each structure whether it is abnormal with a confidence score) by either computing MSE in each of these structures and statistically comparing them to ones from healthy samples, or we could compare the number of detections in each structure to the number of detections we should have under the null hypothesis.

Long-Term Perspectives

On the topic of long-term perspectives, one time-consuming task would be to constitute an even larger database than the one we already gathered. We already collected around 2000 samples, but new database are set up every few months or years, with hundreds (or thousands) of new available samples. A database of 10000 samples, although computationally more expensive to train, could improve the performances of the more complex, non-linear methods. It could also help with covariables such as age or sex by having enough samples to train our models only on samples close to our tests subjects on these parameters (of which we had too few).

A closely related long-term perspective is the use of even more complex non-linear methods, such as the recent trend of deep learning methods. Generative Adversarial Networks have had *impressive* results on texture synthesis and even in some anomaly detection related task (albeit in biological image fields

rather than medical ones) and it is only a question of time (research and GPU computing power are still needed) before a 3D, convolutional GAN is set up to perform on MRI datasets. However, as I did mention in one of the earlier sections, it is my personal belief that these deep learning algorithms will only perform strongly better than ours given a large enough number of samples.

Finally, a very obvious but difficult perspective would be to look for other applications to our methods, as our contributions are mostly methodological: any of our methods could be applied to a dataset of registered images for instance, or to any "machine learning" dataset as long as features "corresponds" from sample to sample. Of course such methods would only be of interest on non-linear datasets, but fortunately they are the most common ones.

Appendix

Manifold

Smoothness: Let $V \in \mathbb{R}^N$, $a \in V$, and $d \in \mathbb{N}$. We say that V is smooth at point a of dimension d if and only if there exists F, a $\mathcal{C}^1$-diffeomorphism from $U \in \mathcal{V}(a) \subset \mathbb{R}^N$ to $F(U) \in \mathcal{V}(0) \subset \mathbb{R}^N$ that transforms V into a d-dimensional vector space. $F(V \cap U) = V' \cap F(U)$, with $V' = \mathbb{R}^d \times \{0\} \in \mathbb{R}^N$

Manifold [93]: V is a d-dimensional manifold if it is smooth in all of its points. A Riemannian manifold is one that includes a differential structure and a Riemannian metric to allow for functional analysis over the manifold.

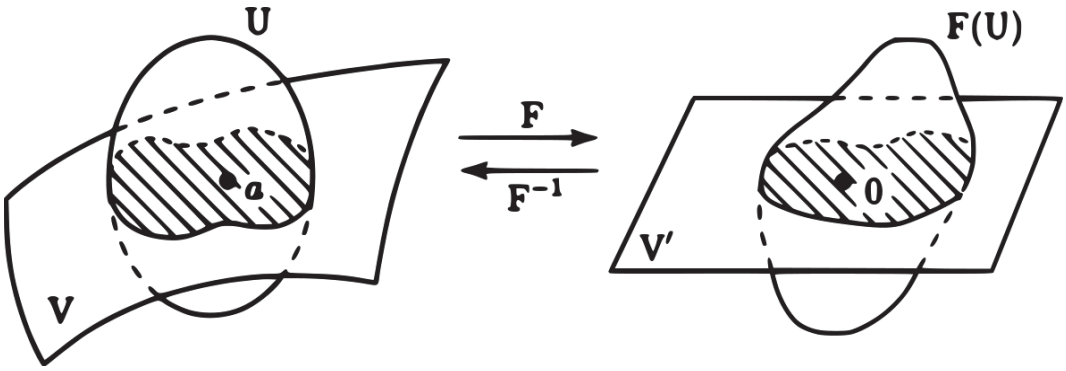

Figure 1: Illustration of a manifold [93]. The diffeomorphism F allows for local areas around manifold points to become linear.

Figure 1 provides an illustration of the definition of a manifold with the previous notations, while figure 2 presents an illustration of the manifold hypothesis and of our paradigm for a dataset of satellite images coming from the same area.

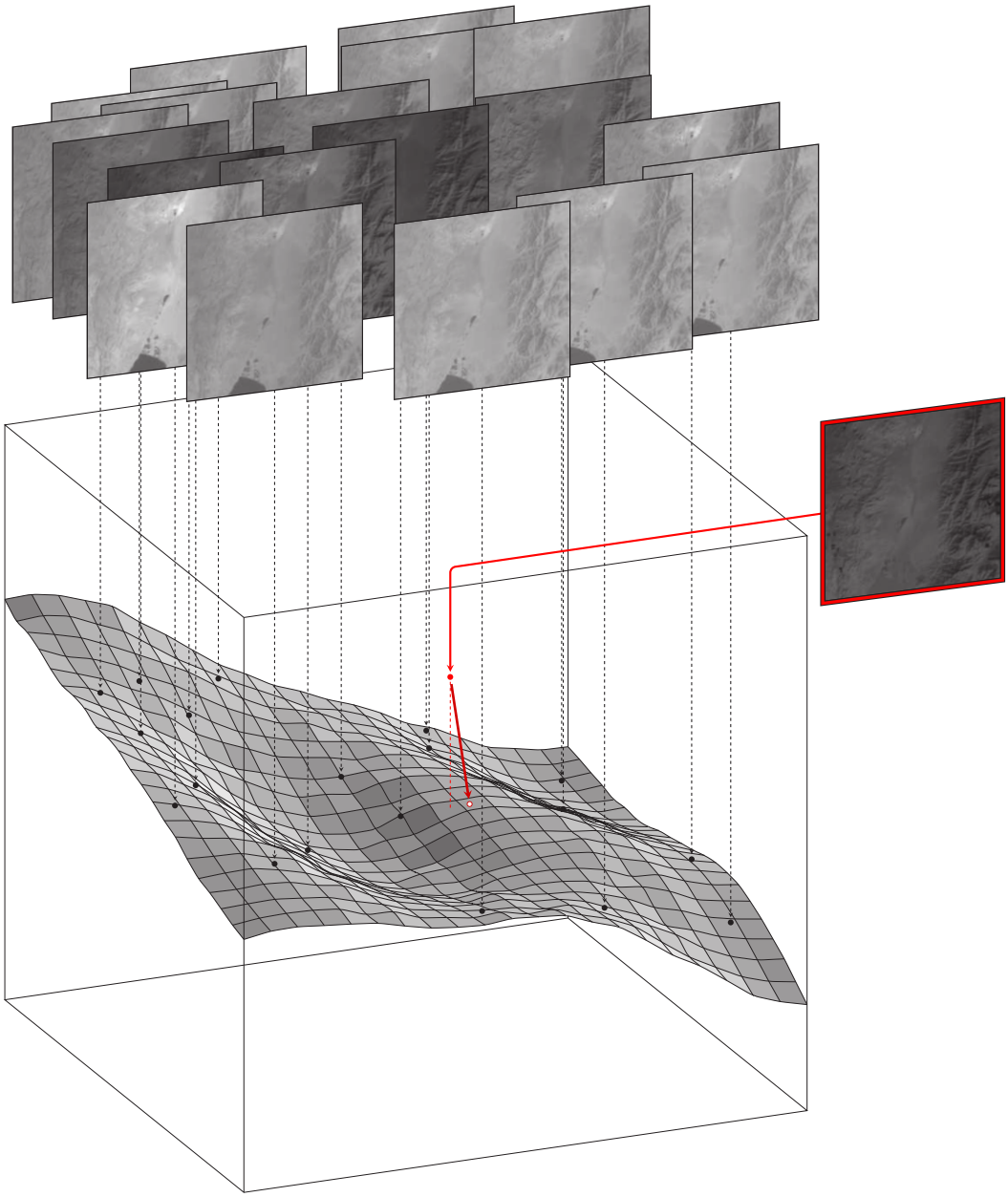

Figure 2: 3D representation of a manifold of satellite images lying in high-dimension, and projection of a new data point over this manifold. Credit goes to V.Vidal.

Publications from the Author

The author has published the following list of international conference or journal articles. Note that only the last one focuses on the topic developed in this work.

[94]: P-H Conze, Florian Tilquin, Vincent Noblet, François Rousseau, Fabrice Heitz, and Patrick Pessaux. Hierarchical multi-scale supervoxel matching using random forests for automatic semi-dense abdominal image registration. In 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), pages 490–493. IEEE, 2017

[95]: Pierre-Henri Conze, Florian Tilquin, Mathieu Lamard, Fabrice Heitz, and Gwenole Quellec. Long-term superpixel tracking using unsupervised learning and multi-step integration. In 2018 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), pages 1–6. IEEE, 2018

[96]: Florian Tilquin, Pierre-Henri Conze, Patrick Pessaux, Mathieu Lamard, Gwenolé Quellec, Vincent Noblet, and Fabrice Heitz. Robust supervoxel matching combining mid-level spectral and context-rich features. In International Workshop on Patch-based Techniques in Medical Imaging, pages 39–47. Springer, 2018

[97]: Hugo Touvron, Sylvain Faisan, Florian Tilquin, and Vincent Noblet. Pitfalls related to computer-aided diagnosis system learned from multiple databases. In 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), pages 806–809. IEEE, 2019

[98]: Pierre-Henri Conze, Florian Tilquin, Mathieu Lamard, Fabrice Heitz, and Gwenolé Quellec. Unsupervised learning-based long-term superpixel tracking. Image and Vision Computing, 89:289 – 301, 2019

[99]: Florian Tilquin, Sylvain Faisan, Fabrice Heitz, Vincent Noblet, Frédéric Blanc, and Izzie Namer. Anomaly detection in single subject vs group using manifold learning. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2867–2871. IEEE, 2019

Bibliography

[1] Torbjørn Vik, Fabrice Heitz, Izzie Namer, and Jean-Paul Armspach. On the modeling, construction, and evaluation of a probabilistic atlas of brain perfusion. NeuroImage, 24(4):1088–1098, 2005.

[2] RB Marimont and MB Shapiro. Nearest neighbour searches and the curse of dimensionality. IMA Journal of Applied Mathematics, 24(1):59–70, 1979.

[3] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In Proceedings of the thirtieth annual ACM symposium on Theory of computing, pages 604–613. ACM, 1998.

[4] Richard E Bellman. Adaptive control processes: a guided tour, volume 2045. Princeton university press, 2015.

[5] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? In International conference on database theory, pages 217–235. Springer, 1999.

[6] Gunnar Carlsson. Topology and data. Bulletin of the American Mathematical Society, 46(2):255–308, 2009.

[7] Karl Pearson. Principal components analysis. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 6(2):559, 1901.

[8] Harold Hotelling. Analysis of a complex of statistical variables into principal components. Journal of educational psychology, 24(6):417, 1933.

[9] Gene H Golub and Christian Reinsch. Singular value decomposition and least squares solutions. In Linear Algebra, pages 134–151. Springer, 1971.

[10] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. Chemometrics and intelligent laboratory systems, 2(1-3):37–52, 1987.

[11] Markus Ringnér. What is principal component analysis? Nature biotechnology, 26(3):303, 2008.

[12] Warren S Torgerson. Theory and methods of scaling. 1958.

[13] Warren S Torgerson. Multidimensional scaling: I. theory and method. Psychometrika, 17(4):401–419, 1952.

[14] Teuvo Kohonen. The self-organizing map. Proceedings of the IEEE, 78(9):1464–1480, 1990.

[15] Teuvo Kohonen. Exploration of very large databases by self-organizing maps. In Proceedings of International Conference on Neural Networks (ICNN'97), volume 1, pages PL1–PL6. IEEE, 1997.

[16] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In International Conference on Artificial Neural Networks, pages 583–588. Springer, 1997.

[17] Sebastian Mika, Bernhard Schölkopf, Alex J Smola, Klaus-Robert Müller, Matthias Scholz, and Gunnar Rätsch. Kernel pca and de-noising in feature spaces. In Advances in neural information processing systems, pages 536–542, 1999.

[18] MA Aizerman. The probability problem of pattern recognition learning and the method of potential functions. Automation and Remote Control, 25:1175–1193, 1964.

[19] Joshua B Tenenbaum. Mapping a manifold of perceptual observations. In Advances in neural information processing systems, pages 682–688, 1998.

[20] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. science, 290(5500):2319–2323, 2000.

[21] Vin D Silva and Joshua B Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. In Advances in neural information processing systems, pages 721–728, 2003.

[22] Edsger W Dijkstra. A note on two problems in connexion with graphs. Numerische mathematik, 1(1):269–271, 1959.

[23] Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika, 29(1):1–27, 1964.

[24] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. science, 290(5500):2323–2326, 2000.

[25] Lawrence K Saul and Sam T Roweis. An introduction to locally linear embedding. unpublished. Available at: http://www. cs. toronto. edu/~ roweis/lle/publications. html, 2000.

[26] Ronald R Coifman, Stephane Lafon, Ann B Lee, Mauro Maggioni, Boaz Nadler, Frederick Warner, and Steven W Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. Proceedings of the National Academy of Sciences of the United States of America, 102(21):7426–7431, 2005.

[27] Ronald R Coifman and Stéphane Lafon. Diffusion maps. Applied and computational harmonic analysis, 21(1):5–30, 2006.

[28] Ronald R Coifman and Stéphane Lafon. Geometric harmonics: a novel tool for multiscale out-of-sample extension of empirical functions. Applied and Computational Harmonic Analysis, 21(1):31–52, 2006.

[29] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Advances in neural information processing systems, pages 585–591, 2002.

[30] David L Donoho and Carrie Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Sciences, 100(10):5591–5596, 2003.

[31] Fan RK Chung and Fan Chung Graham. Spectral graph theory. Number 92. American Mathematical Soc., 1997.

[32] Russell Merris. A note on laplacian graph eigenvalues. Linear algebra and its applications, 285(1-3):33–35, 1998.

[33] Russell Merris. Laplacian graph eigenvectors. Linear algebra and its applications, 278(1-3):221–236, 1998.

[34] Laleh Haghverdi, Florian Buettner, and Fabian J Theis. Diffusion maps for high-dimensional single-cell analysis of differentiation data. Bioinformatics, 31(18):2989–2998, 2015.

[35] Torbjørn Vik, Fabrice Heitz, and Pierre Charbonnier. Robust pose estimation and recognition using non-gaussian modeling of appearance subspaces. IEEE transactions on pattern analysis and machine intelligence, 29(5), 2007.

[36] Christopher TH Baker. The numerical treatment of integral equations. 1977.

[37] Evert Johannes Nyström. Über die praktische Auflösung von linearen Integralgleichungen mit Anwendungen auf Randwertaufgaben der Potentialtheorie. Akademische Buchhandlung, 1929.

[38] Petros Drineas and Michael W Mahoney. On the nyström method for approximating a gram matrix for improved kernel-based learning. journal of machine learning research, 6(Dec):2153–2175, 2005.

[39] Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik. Spectral grouping using the nystrom method. IEEE transactions on pattern analysis and machine intelligence, 26(2):214–225, 2004.

[40] Yoshua Bengio, Jean-françcois Paiement, Pascal Vincent, Olivier Delalleau, Nicolas L Roux, and Marie Ouimet. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. In Advances in neural information processing systems, pages 177–184, 2004.

[41] Carlotta Orsenigo and Carlo Vercellis. Kernel ridge regression for out-of-sample mapping in supervised manifold learning. Expert Systems with Applications, 39(9):7757–7762, 2012.

[42] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. R news, 2(3):18–22, 2002.

[43] Gökhan H Bakir, Jason Weston, and Bernhard Schölkopf. Learning to find pre-images. Advances in neural information processing systems, 16(7):449–456, 2004.

[44] Bernhard Scholkopf, Sebastian Mika, Chris JC Burges, Philipp Knirsch, K-R Muller, Gunnar Ratsch, and Alexander J Smola. Input space versus feature space in kernel-based methods. IEEE transactions on neural networks, 10(5):1000–1017, 1999.

[45] JT-Y Kwok and IW-H Tsang. The pre-image problem in kernel methods. IEEE transactions on neural networks, 15(6):1517–1525, 2004.

[46] Pablo Arias, Gregory Randall, and Guillermo Sapiro. Connecting the out-of-sample and pre-image problems in kernel methods. In 2007 IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8. IEEE, 2007.

[47] Paul Honeine and Cédric Richard. Solving the pre-image problem in kernel machines: A direct method. In 2009 IEEE International Workshop on Machine Learning for Signal Processing, pages 1–6. IEEE, 2009.

[48] Elizbar A Nadaraya. On estimating regression. Theory of Probability & Its Applications, 9(1):141–142, 1964.

[49] Samuel Gerber, Tolga Tasdizen, Sarang Joshi, and Ross Whitaker. On the manifold structure of the space of brain images. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 305–312. Springer, 2009.

[50] Samuel Gerber, Tolga Tasdizen, and Ross Whitaker. Dimensionality reduction and principal surfaces via kernel map manifolds. In 2009 IEEE 12th International Conference on Computer Vision, pages 529–536. IEEE, 2009.

[51] Peter Meinicke, Stefan Klanke, Roland Memisevic, and Helge Ritter. Principal surfaces from unsupervised kernel regression. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(9):1379–1391, 2005.

[52] Patrick Etyngier, Florent Segonne, and Renaud Keriven. Shape priors using manifold learning techniques. In Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, pages 1–8. IEEE, 2007.

[53] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[54] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. Neural computation, 1(4):541–551, 1989.

[55] Nathaniel Rochester, J Holland, L Haibt, and W Duda. Tests on a cell assembly theory of the action of the brain, using a large digital computer. IRE Transactions on information Theory, 2(3):80–93, 1956.

[56] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review, 65(6):386, 1958.

[57] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

[58] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.

[59] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. Neural computation, 18(7):1527–1554, 2006.

[60] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. Restricted boltzmann machines for collaborative filtering. In Proceedings of the 24th international conference on Machine learning, pages 791–798. ACM, 2007.

[61] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.

[62] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434, 2015.

[63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[64] Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. Neural networks, 2(1):53–58, 1989.

[65] Hervé Bourlard and Yves Kamp. Auto-association by multilayer perceptrons and singular value decomposition. Biological cybernetics, 59(4-5):291–294, 1988.

[66] Andrew Ng et al. Sparse autoencoder. CS294A Lecture notes, 72(2011):1–19, 2011.

[67] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

[68] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In International Conference on Artificial Neural Networks, pages 52–59. Springer, 2011.

[69] Wilfrid J Dixon and Frank J Massey Jr. Introduction to statistical analysis. 1951.

[70] Gary Chamberlain. Analysis of covariance with qualitative data, 1979.

[71] James H Bray, Scott E Maxwell, and Scott E Maxwell. Multivariate analysis of variance. Number 54. Sage, 1985.

[72] KJ Friston, CD Frith, PF Liddle, Raymond J Dolan, AA Lammertsma, and RSJ Frackowiak. The relationship between global and local changes in pet scans. Journal of Cerebral Blood Flow & Metabolism, 10(4):458–466, 1990.

[73] Karl J Friston, Andrew P Holmes, Keith J Worsley, J-P Poline, Chris D Frith, and Richard SJ Frackowiak. Statistical parametric maps in functional imaging: a general linear approach. Human brain mapping, 2(4):189–210, 1994.

[74] John Ashburner and Karl J Friston. Voxel-based morphometry—the methods. Neuroimage, 11(6):805–821, 2000.

[75] John Ashburner and Karl J Friston. Why voxel-based morphometry should be used. Neuroimage, 14(6):1238–1243, 2001.

[76] Catherine J Mummery, Karalyn Patterson, Cathy J Price, John Ashburner, Richard SJ Frackowiak, and John R Hodges. A voxel-based morphometry study of semantic dementia: relationship between temporal lobe atrophy and semantic memory. Annals of neurology, 47(1):36–45, 2000.

[77] Yoko Hirata, Hiroshi Matsuda, Kiyotaka Nemoto, Takashi Ohnishi, Kentaro Hirao, Fumio Yamashita, Takashi Asada, Satoshi Iwabuchi, and Hirotsugu Samejima. Voxel-based morphometry to discriminate early alzheimer's disease from controls. Neuroscience letters, 382(3):269–274, 2005.

[78] Patrick Etyngier, Florent Ségonne, and Renaud Keriven. Active-contour-based image segmentation using machine learning techniques. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 891–899. Springer, 2007.

[79] Patrick Etyngier, Renaud Keriven, and Florent Ségonne. Projection onto a shape manifold for image segmentation with prior. In 2007 IEEE International Conference on Image Processing, volume 4, pages IV–361. IEEE, 2007.

[80] Nicolas Thorstensen, Florent Segonne, and Renaud Keriven. Pre-image as karcher mean using diffusion maps: Application to shape and image denoising. Scale Space and Variational Methods in Computer Vision, pages 721–732, 2009.

[81] William S Cleveland. Robust locally weighted regression and smoothing scatterplots. Journal of the American statistical association, 74(368):829–836, 1979.

[82] Gary E Christensen, Richard D Rabbitt, Michael I Miller, et al. Deformable templates using large deformation kinematics. IEEE transactions on image processing, 5(10):1435–1447, 1996.

[83] M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. International journal of computer vision, 61(2):139–157, 2005.

[84] Peter J Huber et al. The behavior of maximum likelihood estimates under nonstandard conditions. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1, pages 221–233. University of California Press, 1967.

[85] Ricardo Antonio Maronna. Robust m-estimators of multivariate location and scatter. The annals of statistics, pages 51–67, 1976.

[86] Peter J Huber. Robust statistics. Wiley Series in Probability and Mathematical Statistics, New York: Wiley,| c1981, 1981.

[87] Paul W Holland and Roy E Welsch. Robust regression using iteratively reweighted least-squares. Communications in Statistics-theory and Methods, 6(9):813–827, 1977.

[88] Gunilla Borgefors. Distance transformations in digital images. Computer vision, graphics, and image processing, 34(3):344–371, 1986.

[89] Clifford R Jack, Matt A Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L Whitwell, Chadwick Ward, et al. The alzheimer's disease neuroimaging initiative (adni): Mri methods. Journal of magnetic resonance imaging, 27(4):685–691, 2008.

[90] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Diffeomorphic demons: Efficient non-parametric image registration. NeuroImage, 45(1):S61–S72, 2009.

[91] Brian B Avants, Nick Tustison, and Gang Song. Advanced normalization tools (ants). Insight j, 2:1–35, 2009.

[92] Mark Jenkinson, Christian F Beckmann, Timothy EJ Behrens, Mark W Woolrich, and Stephen M Smith. Fsl. Neuroimage, 62(2):782–790, 2012.

[93] François Rouvière. Petit guide de calcul différentiel: à l'usage de la licence et de l'agrégation. 2009.

[94] P-H Conze, Florian Tilquin, Vincent Noblet, François Rousseau, Fabrice Heitz, and Patrick Pessaux. Hierarchical multi-scale supervoxel matching using random forests for automatic semi-dense abdominal image registration. In 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), pages 490–493. IEEE, 2017.

[95] Pierre-Henri Conze, Florian Tilquin, Mathieu Lamard, Fabrice Heitz, and Gwenole Quellec. Long-term superpixel tracking using unsupervised learning and multi-step integration. In 2018 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), pages 1–6. IEEE, 2018.

[96] Florian Tilquin, Pierre-Henri Conze, Patrick Pessaux, Mathieu Lamard, Gwenolé Quellec, Vincent Noblet, and Fabrice Heitz. Robust supervoxel matching combining mid-level spectral and context-rich features. In International Workshop on Patch-based Techniques in Medical Imaging, pages 39–47. Springer, 2018.

[97] Hugo Touvron, Sylvain Faisan, Florian Tilquin, and Vincent Noblet. Pitfalls related to computer-aided diagnosis system learned from multiple databases. In 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), pages 806–809. IEEE, 2019.

[98] Pierre-Henri Conze, Florian Tilquin, Mathieu Lamard, Fabrice Heitz, and Gwenolé Quellec. Unsupervised learning-based long-term superpixel tracking. Image and Vision Computing, 89:289 – 301, 2019.

[99] Florian Tilquin, Sylvain Faisan, Fabrice Heitz, Vincent Noblet, Frédéric Blanc, and Izzie Namer. Anomaly detection in single subject vs group using manifold learning. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2867–2871. IEEE, 2019.

[100] Ian T Jolliffe. Principal component analysis and factor analysis. In Principal component analysis, pages 115–128. Springer, 1986.

[101] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 61(3):611–622, 1999.

[102] Michael E Tipping and Christopher M Bishop. Mixtures of probabilistic principal component analyzers. Neural computation, 11(2):443–482, 1999.

[103] Neil D Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. In Advances in neural information processing systems, pages 329–336, 2004.

[104] Christopher M Bishop, Markus Svensén, and Christopher KI Williams. Gtm: The generative topographic mapping. Neural computation, 10(1):215–234, 1998.

[105] Míriam López, Javier Ramírez, Juan Manuel Górriz, Ignacio Álvarez, Diego Salas-Gonzalez, Fermín Segovia, Rosa Chaves, Pablo Padilla, Manuel Gómez-Río, Alzheimer's Disease Neuroimaging Initiative, et al. Principal component analysis-based techniques and supervised classification schemes for the early detection of alzheimer's disease. 74(8):1260–1271, 2011.

[106] Catriona D Good, Ingrid S Johnsrude, John Ashburner, Richard NA Henson, Karl J Friston, and Richard SJ Frackowiak. A voxel-based morphometric study of ageing in 465 normal adult human brains. Neuroimage, 14(1):21–36, 2001.

[107] Mark L Davison. Multidimensional scaling. 1991.

[108] Joseph B Kruskal and Myron Wish. Multidimensional scaling, volume 11. Sage, 1978.

[109] Association Alzheimer's. 2015 alzheimer's disease facts and figures. Alzheimer's & dementia: the journal of the Alzheimer's Association, 11(3):332, 2015.

[110] Daniel S Marcus, Tracy H Wang, Jamie Parker, John G Csernansky, John C Morris, and Randy L Buckner. Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults. Journal of cognitive neuroscience, 19(9):1498–1507, 2007.

[111] Mark W Woolrich, Saad Jbabdi, Brian Patenaude, Michael Chappell, Salima Makni, Timothy Behrens, Christian Beckmann, Mark Jenkinson, and Stephen M Smith. Bayesian analysis of neuroimaging data in fsl. Neuroimage, 45(1):S173–S186, 2009.

[112] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. Radiology, 143(1):29–36, 1982.

[113] Fred L Bookstein. "voxel-based morphometry" should not be used with imperfectly registered images. Neuroimage, 14(6):1454–1462, 2001.

[114] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 1975–1981. IEEE, 2010.

[115] Louis Kratz and Ko Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 1446–1453. IEEE, 2009.

[116] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. ACM computing surveys (CSUR), 41(3):15, 2009.

Modèles statistiques sur des variétés pour la détection d'anomalies dans les images médicales

Introduction

L'objectif de cette thèse est de traiter le problème de détection des anomalies dans les images neurologiques, dans un contexte de comparaison entre un sujet et un groupe contrôle. Bien que la plupart des algorithmes de l'état de l'art de la détection d'anomalies se concentrent sur la détection de données aberrantes par techniques dites "un-contre-N", notre but ici est plutôt de fournir une localisation spatiale des motifs aberrants au sein du sujet testé (s'il y a lieu).

Cette localisation est obtenue en créant un modèle de normalité appris sur les échantillons du groupe contrôle, qui est ensuite appliqué aux images test dans le but d'évaluer si celles-ci se conforment ou non au modèle. Cette tâche peut être effectuée par des experts, mais la taille des images testées et la précision (voxellique) du résultat demandé motivent grandement l'usage d'algorithmes pour les guider dans leur diagnostic.

La figure 1 illustre le type de détection qeu l'on cherche à réaliser sur une image IRM.

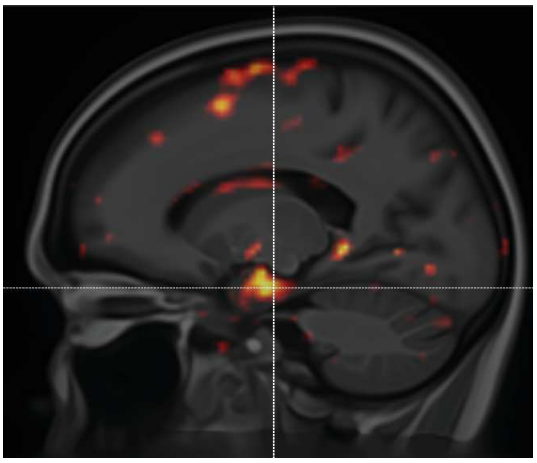

Figure 1: Illustration de la détection d'anomalies obtenue sur une image IRM (coupe sagittale). Les zones en rouge ou jaunes sont détectées comme anormales.

Cette thèse s'appuie principalement sur les travaux d'un précédent doctorant de l'équipe, Vik Torbjørn, et cherche à développer une extension de ses méthodes. Durant sa thèse, il a développé plusieurs méthodes de détection d'anomalies basées sur des modèles linéaires multivariés comme l'Analyse en Composantes Principales (ACP), avec le même type d'applications médicales.

Tout comme à l'époque de Torbjørn, un autre algorithme sera un concurrent notoire de nos méthodes : le modèle linéaire général, dont l'implémentation *Statistical Parametric Mapping* (SPM) est encore aujourd'hui extrêmement répandue dans le milieu médical. Le modèle linéaire général est un modèle univarié et linéaire qui fait de plus des hypothèses fortes sur la distribution des données analysées. La popularité de l'algorithme SPM s'explique par sa capacité à fournir une détection rapide et fiable sur de larges jeux de données, basée sur de solides fondations statistiques.

Les méthodes que nous avons cherché à développer au cours de cette thèse suivent le "cahier des charges" suivant : il s'agit de méthodes non-linéaires, multivariées (exploitant les corrélations spatiales au sein de nos données) ne faisant pas d'hypothèse supplémentaire sur la distribution des échantillons.

Méthodes

Paradigme

La plupart des méthodes de détection élaborées dans cette thèse s'appuient sur un paradigme que l'on peut exprimer comme suit :

Étant donné un jeu de donné X représentant la normalité contre laquelle on souhaite confronter un sujet test Y, la détection d'anomalies se ramène à trouver l'image la plus proche de Y appartenant à la structure géométrique sous-jacente à X.

La "normalisation" d'un sujet test Y s'apparente dans notre méthodologie à trouver une fonction de projection μ dans l'espace image de grande dimension associant à un point test son point le plus proche appartenant à la structure géométrique de X. Il s'agit dans notre cas d'établir un modèle qui soit globalement cohérent avec le jeu de données contrôle. Cette tâche a été effectuée à l'aide d'algorithmes d'apprentissage statistique utilisant le jeu de données contrôle comme données d'entraînement pour l'apprentissage de leur modèle.

Les algorithmes *classiques* d'apprentissage statistique étant malheureusement perturbés par la grande dimensionnalité des données auxquelles on souhaite les appliquer (du à la malédiction de la dimension), nous nous sommes appuyés sur **l'hypothèse de variété** pour compléter ce paradigme. En effet, celle-ci stipule que les images naturelles (et par conséquent les images médicales) ne sont pas distribuées aléatoirement dans l'espace vectoriel ambiant auquel elles appartiennent, mais forment en réalité un sous-espace géométrique structuré, non-linéaire, appelé variété. Ces variétés ont la propriété d'être des espaces

localement linéaires possédant leur propre dimension, bien plus faible que celle de l'espace ambiant.

Une fois la projection (ou normalisation) obtenue, la détection d'anomalie s'effectue par test statistique univarié: le résidu entre un individu test et sa projection est confronté statistiquement en chaque voxel à celui d'individus témoins extraits du groupe contrôle X au préalable. On calcule la variance des résidus contrôle au voxel i : σ_i^2, puis l'on compare le résidu test à cette variance par le test suivant: $T(Y_i) = \frac{\mu(Y_i)-Y_i}{\sigma_i}$.

Réduction de dimension

Les méthodes de réduction de dimension linéaire sont les plus établies de l'état de l'art, l'ACP en étant la principale représentante. Vik Torbjørn a dans ses travaux mis à profit les capacités de l'ACP à compresser l'information pour établir un algorithme de normalisation des données. Sa méthode la plus aboutie modifie la réduction de dimension de l'ACP en remplaçant la projection orthogonale aux moindres carrés par une version pondérée, tout en rajoutant une modélisation de la distribution dans le sous-espace qui contraint d'avantage encore la projection d'un sujet test à respecter celle des projections d'individus témoins.

Les méthodes élaborées durant cette thèse mettent en avant les deux propriétés intrinsèques aux variétés. En effet, dans le but d'obtenir une *fonction de projection* μ qui associe à chaque image test Y sa version "normalisée" (*i.e.* la plus proche de Y pouvant appartenir à la variété des données contrôle -comme motivé par le paradigme utilisé-), nous avons en effet utilisé des algorithmes de réduction de dimension non-linéaires. Ces algorithmes modélisent la géométrie de l'ensemble contrôle X dans une première phase d'apprentissage, en associant à chaque point de X dans l'espace d'origine de grande dimension un point dans un espace de dimension réduite (ou espace réduit), choisie par l'utilisateur (pour correspondre au mieux à celle de la variété de X).

La réduction de dimension sert dans notre méthodologie à effectuer une première étape de "compression" de l'information visant à extraire de nos données à grande dimensionnalité l'information nécessaire à la prise en compte de la structure géométrique de l'espace contrôle. Toutefois, une seconde étape d'extension de la réduction de dimension à n'importe quel point test Y, ainsi qu'une reconstruction de l'image (*i.e.* un passage de la dimension réduite à l'espace image d'origine) sont nécessaires pour obtenir la fonction de projection μ souhaitée (voir la figure de synthèse 2).

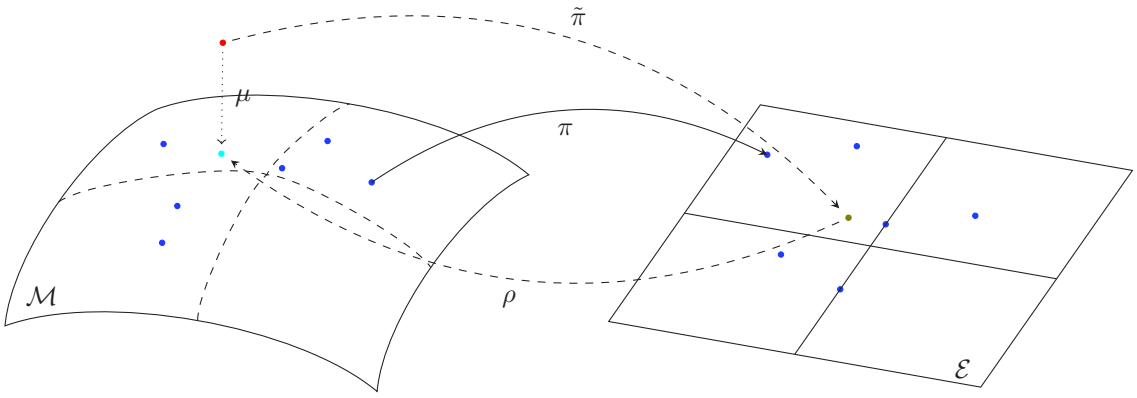

Figure 2: Synthèse de la méthodologie proposée. Les échantillons contrôle (en bleu), sur la variété $\mathcal{M}$ sont "réduits" dans le sous-espace $\mathcal{E}$ par π. Un nouvel échantillon (en rouge) est lui aussi réduit (en vert olive) *via* l'extension $\tilde{\pi}$. Enfin, un point de la variété $\mathcal{M}$ (en cyan) correspondant au point test est trouvé par la reconstruction ρ.

Nous avons sélectionné plusieurs algorithmes de réduction de dimension classiques pour nos besoins: notamment *Isometric feature Mapping* (ISOMAP), et *Locally Linear Embedding* (LLE). ISOMAP est une méthode globale fondé sur la constitution d'un noyau géodésique (longueur du plus court chemin entre deux points de $\mathcal{M}$ restant sur la variété) entre chacun des points de X; la réduction de dimension associée est alors fondée sur le calcul des vecteurs propres du noyau constitué. L'algorithme LLE se base sur la propriété de linéarité locale pour effectuer une réduction de dimension qui respecte les relations (barycentre euclidien) entre un point de l'espace de grande dimension et ses voisins dans l'espace d'origine en les transposant dans un espace de dimension réduite. À la différence de leur contrepartie linéaire l'analyse en composante principale, ces méthodes ne permettent pas, une fois la réduction de dimension π effectuée sur X de l'étendre naturellement à de nouveaux points, ni de reconstruire les points de dimension réduite. Il a donc été nécessaire d'introduire une extension pour ces méthodes, ainsi qu'une méthode de reconstruction.

Extensions

Pour la méthode ISOMAP, nous avons utilisé l'extension de Nyström afin d'étendre notre réduction de dimension à tout point test. La méthode de Nyström est une méthode de l'état de l'art visant à l'origine à trouver des solutions numériques aux équations intégrales. Elle a depuis été étendue aux problèmes d'échantillonnage ou d'approximation de noyaux. Dans notre cas, elle permet d'apporter une extension à la réduction π de l'algorithme ISOMAP en deux étapes : une première consiste à étendre le noyau géodésique K utilisé par ISOMAP, la seconde à formuler la réduction de dimension d'un point test

Y comme une combinaison linéaire de celles des réductions des points de X :

$$\forall k \in [\![1, m]\!], \tilde{\pi}_k(Y) = \frac{1}{\lambda_k} \sum_{i=1}^{N} \pi_k(X_i) \tilde{K}(Y, X_i) \tag{1}$$

Où les λ_i sont les valeurs propres du noyau géodésique K constitué sur X.

Pour la reconstruction ρ, nous avons utilisé une méthode classique de l'état de l'art de régression à noyaux, dite de Nadaraya-Watson. Celle-ci se base sur les relations entre la réduction de dimension d'un point test et celles des points de l'ensemble d'apprentissage X. Affectées par un noyau gaussien, ces relations servent de coefficient d'une combinaison linéaire (et convexe) des points de X pour effectuer la reconstruction :

$$\rho(y) = \frac{\sum_{i=1}^{N} K_g(X_i, y) X_i}{\sum_{i=1}^{N} K_g(X_i, y)} \tag{2}$$

Avec

$$K_g(X_i, y) = \exp\left(-\frac{\|\pi(X_i) - y\|^2}{\sigma}\right) \tag{3}$$

Dans le cas de LLE, une solution combinant à la fois l'extension et la reconstruction a été proposée dans la liste des contributions, énumérées dans la section suivante.

Contributions

Outre le fait d'utiliser une méthode de projection associant la réduction de dimension d'ISOMAP, l'extension de Nyström et la reconstruction par régression de Nadaraya-Watson, qui n'a à notre connaissance pas été préalablement développé dans l'état de l'art (et encore moins dans un contexte de détection d'anomalies), nous avons développé plusieurs méthodes originales dans cette thèse.

Locally Linear Projection

La première concerne l'extension de LLE : pour cette méthode, il nous a paru plus adapté, plutôt qu'effectuer le même processus que pour ISOMAP (bien que cela eut été possible), d'exploiter les spécificités de LLE pour répondre à nos besoins. Pour obtenir sa réduction de dimension, LLE calcule des "coordonnées barycentrique" dans l'espace d'origine qui sont alors reproduites au mieux dans l'espace réduit. Ces coordonnées sont exploitables pour effectuer une projection directe dans l'espace d'origine : ce sont les coordonnées optimales (au sens d'un critère L_2) d'une projection de l'individu test sur la "base" de ses voisins dans l'ensemble d'apprentissage. Cette projection, par la propriété de linéarité locale des variétés, correspond bien à celle que l'on souhaite obtenir dans notre paradigme de normalisation de l'individu test.

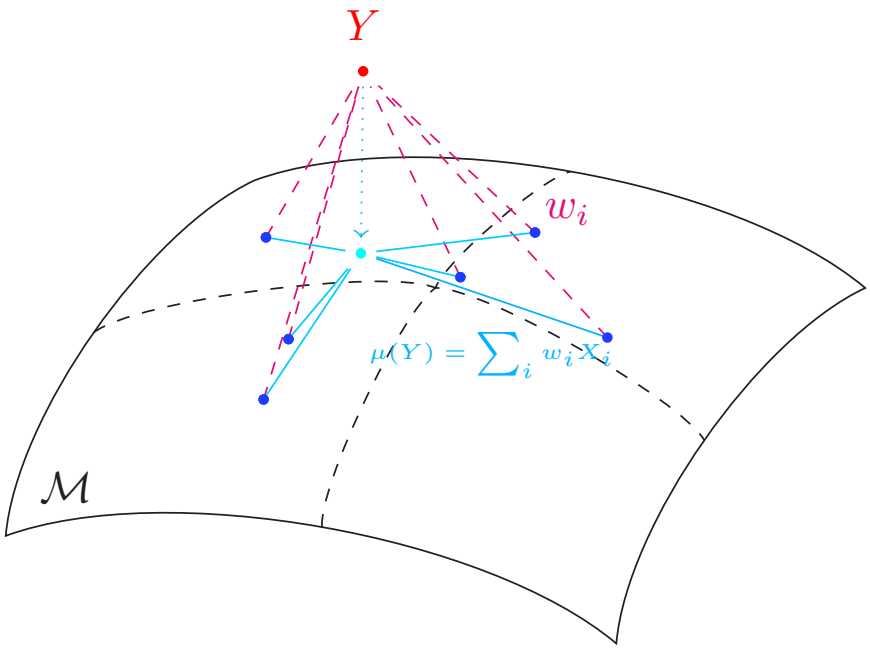

Figure 3: Une représentation de la projection basée sur LLE : un point test (en rouge) est projeté sur $\mathcal{M}$ en utilisant les poids w_i appris sur ses voisins dans l'ensemble d'apprentissage (points bleus). La projection en résultant est illustrée en cyan.

Ainsi, notre projection s'affranchit de la réduction de dimension calculée par LLE, et de ce fait ne nécessite pas non plus de reconstruction depuis l'espace réduit. Cette méthode de projection par la suite été baptisée *Locally Linear Projection* (LLP).

Robustesse

Une autre part importante de nos contributions traite le problème de la robustesse de nos méthodes de projection. En effet dans le cadre de la détection d'anomalies dans des images, nos méthodes sont amenées à devoir traiter des images de sujets pathologiques, qui n'appartiennent donc pas à la variété des contrôles $\mathcal{M}$. Dans ce cas, la pathologie peut profondément modifier l'image test originale (saine), modifiant les relations entre celle-ci et les échantillons contrôle par rapport aux relations originales. Ces relations étant au cœur de nos méthodes de projection, il a fallu développer un moyen de les rendre robustes aux potentielles perturbations induites par les pathologies.

Pour rendre nos méthodes non-linéaires robustes aux anomalies, nous nous sommes inspirés des extensions robustes existantes pour les méthodes linéaires. Celles- ci visent à résoudre une formulation robuste du problème d'optimisation du maximum de vraisemblance, mettant en jeu une fonction de coût pénalisant moins fortement les voxels pour lesquels la reconstruction ne suit pas la

distribution des reconstructions effectuées sur des contrôles. Dans le cas de l'ACP, une solution analytique à ce problème d'optimisation existe et permet d'obtenir une solution robuste d'une manière analogue à la solution classique. Pour les méthodes dérivées de l'ACP rajoutant une modélisation dans le sous-espace telle que celle développée par Torbjørn, une solution analytique n'existe plus. Dans ce cas, des méthodes itératives (IRLS) sont utilisées pour résoudre l'optimisation du maximum de vraisemblance associé.

La solution du problème d'optimisation non-convexe associé au maximum de vraisemblance robuste suivant:

$$\tilde{\pi}_{robust}(Y_n) = \arg\max_{x\in\mathbb{R}^m} \exp\left(-\frac{1}{2}\sum_s f\left(\frac{(Y_n)_s - \rho_{PCA}(x)_s}{\sigma_{LS}}\right)\right) \tag{4}$$

Mettant en jeu la fonction robuste f et la variance obtenue par la reconstruction des moindres carrées σ^2_{LS}, implique de résoudre à chaque étape de l'IRLS le sous-problème:

$$\tilde{\pi}(Y) = \arg\min_y \|B(Y - \overline{X} - Wy)\|_2^2 \tag{5}$$

Dont la solution est analytique.

Bien que nos méthodes n'aient pas de problème d'optimisation simple à mettre en évidence, et ne soient pas directement reliées à un maximum de vraisemblance, nous nous sommes inspirés de cette solution itérative pour résoudre notre problème de robustesse. Dans nos méthodes, la détection d'anomalies basée sur la reconstruction effectuée à une étape permet de calculer des poids pondérant l'influence des voxels de l'individu test lors de la reconstruction de l'étape suivante (la reconstruction initiale étant celle de notre méthode non-robuste) avec itération jusqu'à convergence.

Autres contributions

D'autres contributions sont mises en avant dans ces travaux, mais ont eu moins de portée ou ont été plus difficiles à exploiter pour obtenir des résultats sur données réelles.

Une partie se concentre sur l'utilisation d'une autre méthode de réduction de dimension, appelée Diffusion Maps (DM). Cette technique apporte une modélisation intéressante par rapport à ISOMAP et permet d'obtenir une reconstruction par une méthode d'optimisation, plus spécifique à la méthode de réduction de dimension que la régression à noyaux. Cependant cette méthode de reconstruction étant très coûteuse en temps de calcul, elle a dû être abandonnée sur données réelles.

Une autre contribution originale est une méthode adaptée de la projection obtenue par ISOMAP. Plutôt que d'obtenir l'extension de réduction de dimension par l'extension de Nyström, on l'obtient ici par une optimisation. La reconstruction étant toujours assurée par la régression à noyaux, l'extension d'un point test Y est celle garantissant une reconstruction par Nadaraya-Watson

aussi proche de Y que possible (au sens d'un critère L_2). Cette méthode est malheureusement elle- aussi très coûteuse en temps de calcul.

Résultats

Les résultats obtenus lors de cette thèse sont séparés en deux catégories: ceux obtenus sur données synthétiques et ceux sur données réelles. En effet, bien qu'étant au cœur de l'application de nos méthodes et les plus intéressantes d'un point de vue scientifique, les données réelles ne disposent pas d'une vérité terrain, à laquelle nous pourrions confronter les résultats de nos algorithmes. Nous avons donc du nous rabattre dans un premier temps sur l'élaboration de données synthétiques afin de tester les performances des méthodes développées face aux méthodes préexistantes (SPM et méthode deTorbjørn).

Données synthétiques

Pour élaborer des données synthétiques, nous avons créé des jeux de données non-linéaires, dans lesquels nous contrôlions la distribution et les différents paramètres importants pour nos algorithmes : nombre d'échantillons, nombre de dimensions de la variété intrinsèque aux données, et pourcentage d'anomalie introduite chez les individus tests.

Le premier jeu de données élaboré de ce type consistait à échantilloner des points sur une demi- sphère de dimension plus ou moins grande, et à introduire une anomalie additive représentant un multiple de l'écart-type de la distribution des individus contrôles dans certaines composantes des individus tests. Toutes les méthodes (excepté SPM) ont eu des performances excellentes sur ce jeu de données, linéaires comme non-linéaires. Dès lors, nous avons créé une succession de données synthétiques, à la complexité et au réalisme croissants, afin de se rapprocher au plus des données réelles et avoir une idée des résultats qu'il serait possible d'obtenir dessus.

De points sur une demi-sphère, nous nous sommes tournées vers des images de trapèzes (contrôlées par 3 paramètres) de taille 40 par 40 dans lesquelles une partie de l'image était mise en hyper- intensité chez les données test, puis nous avons utilisé le jeu de données réelles afin de constituer des données synthétiques à partir de véritables IRM. Seules les IRM de sujets témoins étaient alors utilisées, et des sujets tests étaient créés par ajout d'une perturbation dans certaines zones bien choisies.

Les données synthétiques ont permis de confirmer l'intérêt des méthodes non-linéaires développées durant cette thèse sur des jeux de données non-linéaires : en effet, les différents tests réalisés en détection d'anomalies montrent que nos méthodes sont capables d'égaler, voir de surpasser les méthodes linéaires sur ces données. Des tests de robustesse mettent aussi en évidence le fort intérêt des extensions robustes développées lors de cette thèse, dont le gain s'accroît avec l'ampleur de la pathologie .

Données réelles

Bien que ne disposant pas d'une vérité terrain permettant une analyse quantitative des résultats, les données réelles sont exploitables pour une analyse visuelle qualitative avec l'aide d'experts médicaux. Au cours de cette thèse nous avons constitué une base de données regroupant plus de 2000 IRM provenant de plusieurs bases de données internationales ainsi qu'une base de données locale. Environ 200 de ces IRM ont été diagnostiquées par les experts comme souffrant de la maladie d'Alzheimer (les autres étant classifiées comme "témoin"). Ces IRM ont toutes été recalées sur un template commun, et le log-jacobien du champ de déformation issu du recalage entre chaque IRM et le template a été calculé. La base de données finalement utilisée dans la détection d'anomalies est ainsi celle des log-jacobiens de chaque individu, permettant une analyse de tous les sujets dans un espace commun ne mettant en jeu que les déformations de volumes des structures (l'effet principal de la maladie d'Alzheimer étant une régression volumique de certaines zones du cerveau).

L'analyse des résultats de nos méthodes sur données réelles est très satisfaisante. Celles-ci sont capables de pointer dans les jacobiens les zones qui sont le plus classiquement liées à la pathologie par les médecins. Vis-à-vis des méthodes linéaires concurrentes, nos méthodes offent un taux de détection intra-sujets plus important dans les organes les plus souvent atteints par la maladie, ou en détectent certaines que les méthodes linéaires ne détectent pas.

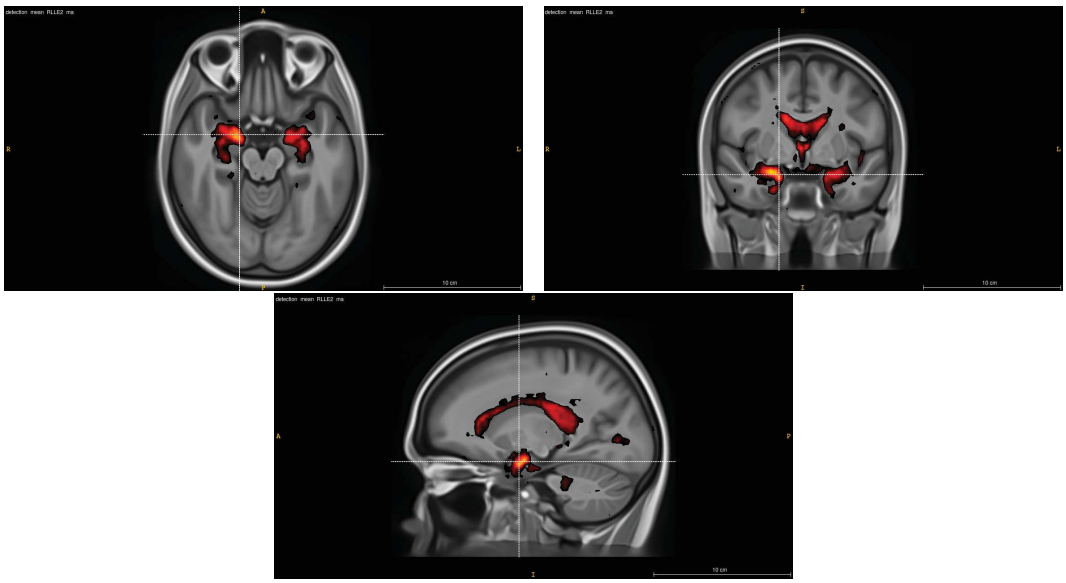

Figure 4: Taux de détection moyen (au sein des sujets) sur la base des données réelles pour l'algorithme de projection par LLE robuste. le taux de détection es seuillé pour ne pas apparaître en dessous de 15% et apparaître blanc au dessus de 50%.

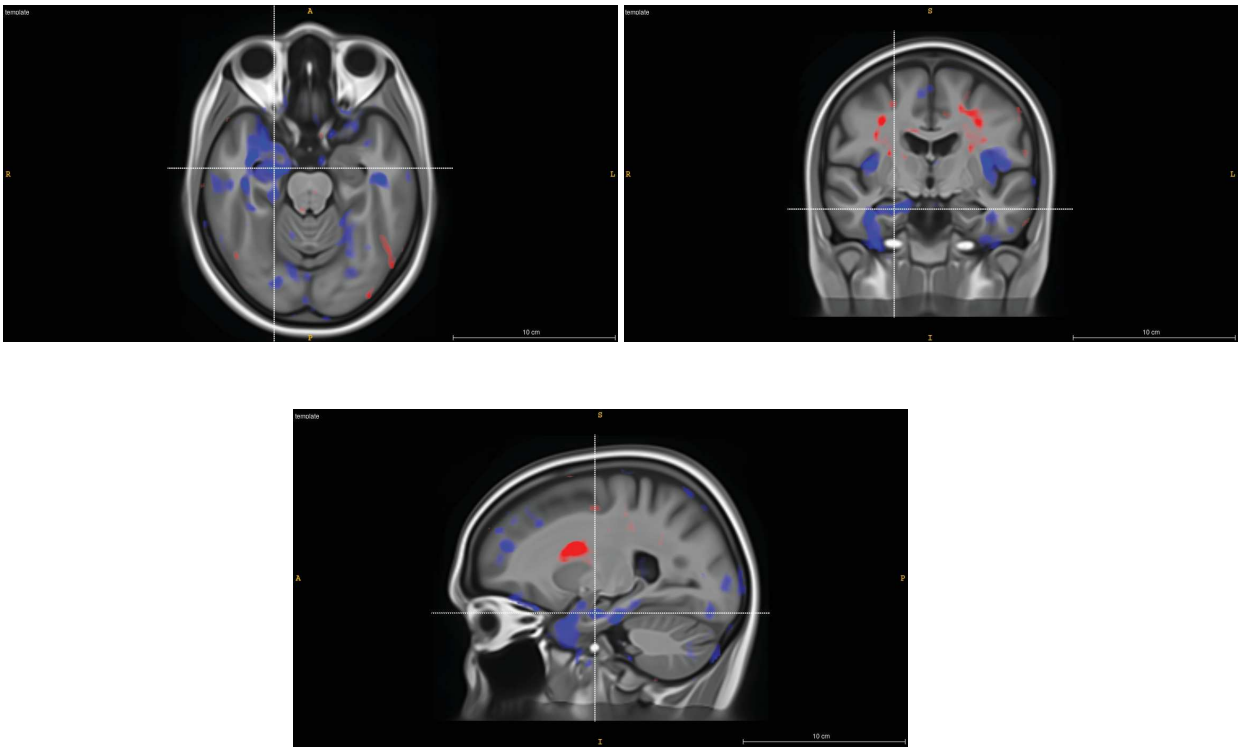

Figure 5: Résultat individuel obtenu par la méthode de projection par LLE robuste. Les dilatations de volumes sont superposées en rouge, et les régressions en bleu.

Conclusion

Cette thèse a permis l'élaboration de plusieurs méthodes originales de détection d'anomalies multivariées et non-linéaires basées sur un paradigme et des techniques de réduction de dimension. Ces méthodes se sont de plus vues augmentées d'une extension robuste cruciale pour traiter des données déviant de la distribution des individus témoin du fait de leur pathologie. Nous avons pu tester extensivement ces nouvelles méthodes sur différents jeux de données synthétiques non-linéaires de notre conception, où l'avantage des méthodes non-linéaires s'est rapidement fait sentir, ainsi que sur un jeu de données médical lui aussi assemblé par nos soins et expertisé par un médecin, sur lequel nos méthodes ont eu de bonnes performances face à leurs contreparties linéaires.

Les suites possibles de ces travaux sont multiples : introduire les différentes covariables liées aux données (âge, sexe, base, etc.), regrouper méthodes linéaires et non-linéaires dans des techniques ensemblistes, ou encore réduire la difficulté du problème en augmentant l'échelle à laquelle se fait la détection (passer des voxels aux organes par exemple), dans un premier temps. Dans un second temps, explorer la piste des techniques d'apprentissage profond très prometteuses, étendre d'avantage notre base de données déjà conséquente, ou bien utiliser nos méthodes sur d'autres bases ayant d'autres applications.