HAL
open science

On metric and statistical properties of topological descriptors for geometric data

Mathieu Carriere

To cite this version:

Mathieu Carriere. On metric and statistical properties of topological descriptors for geometric data. Computational Geometry [cs.CG]. Université Paris 11, 2017. English. NNT : . tel-01659347v1

HAL Id: tel-01659347

https://hal.science/tel-01659347v1

Submitted on 8 Dec 2017 (v1), last revised 30 Jan 2018 (v2)

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

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

NNT : 2017SACLS433

Thèse de doctorat de l'Université Paris-Saclay préparée à l'Université Paris-Sud

Ecole doctorale n°580
Sciences et technologies de l'information et de la communication
Spécialité de doctorat: Informatique
par

M. Mathieu Carrière

On metric and statistical properties of topological descriptors for geometric data

Thèse présentée et soutenue à Paris, le 21 novembre 2017.
Composition du Jury :

M.	Marc Schoenauer,	INRIA Saclay	Président
M.	Gunnar Carlsson,	Université de Stanford	Rapporteur
M.	Jean-Philippe Vert,	Mines ParisTech	Rapporteur
M.	Julien Mairal,	INRIA Grenoble	Rapporteur
M.	Ulrich Bauer,	Université Technique de Munich	Examinateur
M.	Xavier Goaoc,	Université Paris-Est	Examinateur
M.	Steve Oudot,	INRIA Saclay	Directeur de thèse

REMERCIEMENTS

Tout d'abord, j'aimerais remercier les personnes qui ont accepté de faire partie de mon jury de thèse : Julien Mairal, Jean-Philippe Vert, Gunnar Carlsson (tous trois ayant de plus rapporté ce manuscrit), Ulrich Bauer, Marc Schoenauer et Xavier Goaoc. Merci aussi à Tamy Boubekeur pour m'avoir aidé à organiser la soutenance dans les locaux de Telecom Paris Tech. Ce travail a été financé sur la bourse ERC Gudhi (ERC-2013-ADG-339025), obtenue par Jean-Daniel Boissonnat, que je remercie également.

Bien évidemment, la synthèse de ces trois années de travail au sein de l'équipe Geometrica/DataShape (/Tagada?), présentée dans ce document, est moins le fruit d'un travail solitaire que de nombreuses collaborations. A ce titre, je souhaite manifester ma gratitude, pour nos discussions toujours enrichissantes, envers mes coauteurs Maks Ovsjanikov, Bertrand Michel, Marco Cuturi, Ulrich Bauer et tout particulièrement mon directeur de thèse Steve Oudot, dont la disponibilité, la patience, la rigueur et les regards profonds (même lors d'un footing) sur nos sujets d'études ont été des facteurs déterminants pour l'épanouissement de ces trois années de travail, et, à titre plus personnel, pour le plaisir que j'ai eu à travailler pendant ces trois années à Inria.

Ce plaisir est redevable aussi aux membres passés et présents de l'équipe (ainsi que des équipes adjacentes). Je remercie Mickaël, Thomas, Amélie, Alice et Etienne, qui m'ont chaleureusement accueilli, Eddie, avec qui j'ai partagé mon bureau pendant ces trois années, ainsi que les doctorants actuels Dorian, Jérémy, Claire, Nicolas, Théo, Vincent et Raphaël, avec qui j'ai passé de très bons moments au dedans et en dehors du labo, et pour qui je souhaite le meilleur pour les années de recherche à venir. Merci aussi à Marc et Fred, et aux postdocs qui se sont succédés pendant ces trois ans, à savoir Hélène, Clément, Ilaria, Pawel et Miro, pour avoir contribué à cette ambiance amicale au travers de nombreuses discussions. Enfin, pour leur soutien administratif exemplaire, merci à Christine et Stéphanie.

Les deux mois que j'ai passés à Munich dans le groupe Géométrie et Visualisation ont été très enrichissants. Merci à Ulrich Bauer pour avoir permis d'organiser ce séjour, ainsi qu'aux doctorants du groupe pour le chaleureux accueil et les fréquentes séances de bloc.

En dehors du labo, mes amis proches et ma famille ont largement contribué à la qualité de ces trois années. Je souhaite remercier Mathieu, Hugo (Cayla), Hugo (Magaldi), Pauline et Laure, ainsi que tous mes amis parisiens, pour tous les joyeux moments passés ensemble qui ont beaucoup compté pour moi. De même, mes fréquents retours à Toulouse ont à chaque fois été grandement revitalisants grâce à Romain, Pierre, Clément et tous mes amis toulousains de longue date, dont l'amitié m'est très chère. Merci aussi à mes fantastiques colocs Charlotte et Sophie pour tous nos repas et soirées réginaburgiennes que j'ai beaucoup appréciées.

Je remercie toute ma famille (mes cousins et mes cousines, mes grand parents, oncles et tantes), et plus particulièrement, pour leur soutien et leur affection indéfectibles, je remercie mes parents et mes petits frères, de tout mon coeur.

Enfin, merci à toi Aisling, pour tout le temps que nous avons passé ensemble.

CONTENTS

1 Introduction **7**
1.1 Introduction en français . . . 7
1.1.1 Analyse de donnée et apprentissage automatique . . . 7
1.1.2 Descripteurs topologiques . . . 8
1.1.3 Principales limitations . . . 12
1.1.4 Contributions . . . 15
1.2 Introduction in english . . . 17
1.2.1 Data Analysis and Machine Learning . . . 17
1.2.2 Topological Descriptors . . . 17
1.2.3 Main bottlenecks . . . 21
1.2.4 Contributions . . . 24

2 Background on Topology **27**
2.1 Homology Theory . . . 27
2.1.1 Simplices and Simplicial Complexes . . . 27
2.1.2 Simplicial Homology . . . 29
2.1.3 Singular Homology . . . 31
2.1.4 Relative Homology . . . 32
2.2 Persistence Theory . . . 33
2.2.1 Filtrations . . . 33
2.2.2 Persistence Modules . . . 34
2.2.3 Persistence Diagram . . . 35
2.2.4 Stability Properties of Persistence Diagrams . . . 36
2.3 Extended and Levelset Zigzag Persistence . . . 39
2.3.1 Extended persistence . . . 39
2.3.2 Levelset zigzag persistence . . . 41
2.4 Reeb graphs . . . 43
2.4.1 Persistence-based bag-of-features signature . . . 44
2.4.2 Metrics between Reeb graphs . . . 46
2.4.3 Simplification techniques . . . 48
2.4.4 Computation . . . 49
2.5 Mapper . . . 50

3 Telescopes and Reeb graphs **53**
3.1 Telescopes and Operators . . . 54
3.2 A lower bound on d_{b} . . . 60
3.3 Induced Metrics . . . 65
3.4 Conclusion . . . 67

4 Structure and Stability of the Mapper **69**
4.1 Mappers for scalar-valued functions . . . 70
4.2 MultiNerve Mapper . . . 70
4.3 Structure of the MultiNerve Mapper . . . 72
4.3.1 Topological structure of the MultiNerve Mapper . . . 72
4.3.2 A signature for MultiNerve Mapper . . . 75
4.3.3 Induced signature for Mapper . . . 77
4.4 Stability in the bottleneck distance . . . 78
4.5 Stability with respect to perturbations of the cover . . . 80
4.6 Convergence in the functional distortion distance . . . 81
4.6.1 Operators on MultiNerve Mapper . . . 81
4.6.2 Connection between the (MultiNerve) Mapper and the Reeb graph. . . . 82
4.6.3 Convergence results. . . . 85
4.6.4 An alternative proof of Theorem 4.3.3 . . . 86
4.7 Conclusion . . . 88

5 Statistical Analysis and Parameter Selection **89**
5.1 Approximations of (MultiNerve) Mappers and Reeb graphs . . . 89
5.1.1 Approximation tools . . . 90
5.1.2 Discrete approximations . . . 91
5.1.3 Relationships between the constructions . . . 92
5.1.4 Relationships between the signatures . . . 94
5.2 Approximation of a Reeb graph with Mapper . . . 96
5.3 Statistical Analysis of the Mapper . . . 98
5.3.1 Statistical Model for the Mapper . . . 99
5.3.2 Reeb graph inference with exact filter . . . 100
5.3.3 Reeb graph inference with estimated filter . . . 105
5.4 Confidence sets for the signatures . . . 106
5.4.1 Confidence sets . . . 106
5.4.2 Confidence sets derived from Theorem 5.2.1 . . . 107
5.4.3 Bottleneck Bootstrap . . . 108
5.5 Numerical experiments . . . 109
5.5.1 Mappers and confidence regions . . . 109
5.5.2 Noisy data . . . 111
5.6 Conclusion . . . 112

6 Kernel Methods for Persistence Diagrams **113**
6.1 Supervised Machine Learning . . . 114
6.1.1 Empirical Risk Minimization . . . 114
6.1.2 Reproducing Kernel Hilbert Space . . . 115
6.2 A Gaussian Kernel for Persistence Diagrams . . . 116
6.2.1 Wasserstein distance for unnormalized measures on $\mathbb{R}$ . . . 117
6.2.2 The Sliced Wasserstein Kernel . . . 118
6.2.3 Metric Preservation . . . 119
6.2.4 Computation . . . 122
6.2.5 Experiments . . . 122
6.3 Vectorization of Persistence Diagrams . . . 129
6.3.1 Mapping Persistence Diagrams to Euclidean vectors . . . 129
6.3.2 Stability of the topological vectors. . . . 130
6.3.3 Application to 3D shape processing . . . 132
6.4 Conclusion . . . 137

7 Conclusion **139**

A Proof of Lemma 3.4.5 **141**

LIST OF FIGURES

1.1 Déformations du cercle. 8
1.2 Ce nuage de points semble échantillonné sur neuf cercle à petite échelle, et sur un seul cercle à plus grande échelle. 8
1.3 Une base de données d'images. 9
1.4 Diagrames de persistance induit par des boules grossissantes 9
1.5 Diagramme de persistance d'une image 11
1.6 Mapper calculé sur des images 12
1.7 Instabilité de Mappers calculés sur des espaces proches 13
1.8 Instabilité de Mappers calculés avec des couvertures proches 13
1.9 Mapper vu comme une pixelisation du graphe de Reeb 14
1.10 Plan de la thèse 16
1.11 Deformations of a circle. 18
1.12 This point cloud seems to be sampled on nine circles from a small scale, and on a single circle from a larger scale. 18
1.13 A dataset of images. 18
1.14 Persistence diagrams induced by growing balls 19
1.15 Persistence diagram of image 20
1.16 Mapper on images 21
1.17 Instability of Mapper computed on nearby spaces 22
1.18 Instability of Mapper computed with close covers 23
1.19 Mapper as a pixelization of the Reeb graph 23
1.20 Plan of the thesis 25

2.1 Geometric simplices 28
2.2 Geometric simplicial complex 28
2.3 Boundary operator 29
2.4 Cycles 29
2.5 Homology of annulus 30
2.6 Singular simplex 31
2.7 Relative cycle 33
2.8 Lower-star filtration 34
2.9 Persistence diagram induced by filtration 36
2.10 Commutative diagrams for interleaving. 37
2.11 Extended filtration 40
2.12 Mayer-Vietoris half-pyramid 43
2.13 Reeb graph on torus 44
2.14 Two Reeb graphs with the same set of features but not the same layout. 45
2.15 Feature simplification 49
2.16 Mapper on double torus 50

3.1 Merge . . . 56
3.2 Persistence measure for Merge . . . 57
3.3 Split . . . 58
3.4 Up- and down-forks . . . 58
3.5 Shift . . . 60
3.6 Persistence measure for Shift . . . 60
3.7 Simplification operator . . . 61
3.8 Continuous maps . . . 63
3.9 Arc number argument . . . 64
3.10 Branching argument . . . 65
3.11 The space of Reeb graphs is not Cauchy . . . 67

4.1 Simplicial poset . . . 71
4.2 (MultiNerve) Mapper with bivariate map . . . 72
4.3 Left: Staircases of ordinary (light grey) and relative (dark grey) types. Right: Staircases of extended types—$Q^{\mathcal{I}}_{E^-}$ is in dark grey while $Q^{\mathcal{I}}_{E}$ is the union of $Q^{\mathcal{I}}_{E^-}$ with the light grey area. . . . 73
4.4 Pyramid rules . . . 75
4.5 Zigzag persistence modules in the half-pyramid . . . 76
4.6 Mapper as a pixelization of the Reeb graph . . . 77
4.7 Stability of the Mapper . . . 79
4.8 Full transformation on spaces . . . 83
4.9 Full transformation on persistence diagrams . . . 86

5.1 Functions on Mapper . . . 93
5.2 Interval- and intersection-crossing edges . . . 95
5.3 Automatic Mappers on smooth datasets . . . 110
5.4 Automatic Mappers on real-world datasets . . . 111
5.5 Automatic Mappers on a noisy dataset . . . 112

6.1 Kernel trick . . . 116
6.2 Concavity argument . . . 120
6.3 Orbit recognition . . . 125
6.4 Texture and 3D point classification . . . 126
6.5 Accuracy and training time dependences on direction number . . . 127
6.6 Metric distortion . . . 128
6.7 Mapping of a persistence diagram to a sequence with finite support. . . . 129
6.8 Distances to diagonal . . . 130
6.9 Geodesic balls . . . 133
6.10 Geodesic balls . . . 133
6.11 MDS on topological vectors . . . 134
6.12 kNN on topological vectors . . . 135
6.13 Stability of topological vectors . . . 135
6.14 Symmetry . . . 136
6.15 Improvements measured with functional maps . . . 138
6.16 Improvements measured directly on shapes . . . 138

A.1 Images of paths in Reeb graph . . . 142

CHAPTER 1

INTRODUCTION

1.1 Introduction en français

1.1.1 Analyse de donnée et apprentissage automatique

La génération et l'accumulation de données dans des secteurs d'activités variés, autant industriels qu'académiques, ont pris beaucoup d'importance au cours des dernières années, et sont maintenant omniprésents dans de nombreux domaines scientifiques, financiers et industriels. A titre d'exemple, en science du numérique, le développement rapide des processus d'acquisition et de traitement d'images ont permis la mise à disposition publique en ligne d'importantes bases de données [93, 89, 107, 112, 123]. De la même manière, en biologie, la nouvelle génération de séquenceurs ont permis à la plupart des laboratoires d'aisément déterminer l'ADN de différents organismes [14, 78, 88, 100]. Ainsi, la synthétisation et l'extraction d'informations utiles à partir de ces bases de données massives sont devenus des problèmes d'intérêt majeur.

L'apprentissage automatique est un domaine de la science des données dont le but est de fournir des algorithmes ("automatique") pouvant réaliser des prédictions sur de nouvelles données à partir seulement de l'information déjà présente dans des données préalablement collectées ("apprentissage"). Ces techniques permettent de répondre à de multiples problèmes de l'analyse de données, tels que la *classification*, où l'on cherche à prédire des labels, le *clustering*, où l'on cherche à regrouper les données en différents groupes, ou la *régression*, où l'on cherche à approcher une fonction à partir de sa valeur sur les points de données. Nous orientons le lecteur désireux de trouver plus de détails vers [72] pour une introduction complète de ces problématiques. Par exemple, un problème typique de classification est la prédiction de la présence ou non d'effets d'un médicament sur un patient P. Il s'agit d'un problème de classification binaire en cela que les labels à prédire sont au nombre de deux, à savoir "effet" ou "sans effet". En supposant qu'une base de données est disponible, dans laquelle sont enregistrés les effets ou non du médicament sur plusieurs patients, une des manières les plus simples de procéder est de chercher le patient le plus proche de P dans la base de données, et d'attribuer à P le label de ce patient. Cette méthode, simple quoique très efficace, s'appelle la prédiction par le plus proche voisin, et a déjà été étudiée en détail. Plus généralement, la prédiction par le plus proche voisin n'est qu'une méthode parmi de nombreuses autres en apprentissage automatique, qui peuvent traiter de problèmes aussi variés que la classification d'images, la prédiction du genre musical ou le diagnostic médical, pour ne citer que quelques exemples. D'autres exemples d'applications sont présentés dans [72].

Descripteurs. En général, les données prennent la forme de nuage de points dans $\mathbb{R}^D$, où $D \in \mathbb{N}^*$. Chaque point de donnée représente une *observation*, et chaque dimension, ou coordonnée, représente une *mesure*. Par exemple, les observations peuvent être des patients, des images ou des séquences d'ADN, dont les mesures correspondantes seraient des caractéristiques physiques (la taille, le poids, l'âge...), le niveau de gris des pixels, ou des bases azotées A, C, T ou G composant l'ADN. Très souvent, le nombre

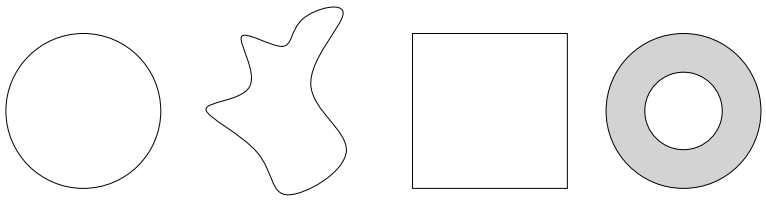

Figure 1.1: Déformations du cercle.

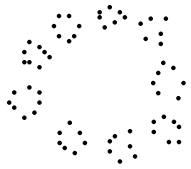

Figure 1.2: Ce nuage de points semble échantillonné sur neuf cercle à petite échelle, et sur un seul cercle à plus grande échelle.

de mesures est élevé, fournissant ainsi beaucoup d'informations, mais rendant dans le même temps les données impossibles à visualiser.

Ainsi, une grande partie de l'analyse de données se consacre à la synthétisation de l'information contenue dans les données en des *descripteurs* simples et interprétables, qui dépendent en général de l'application. Par exemple, on peut touver, parmi les descripteurs usuels : le modèle sac-de-mots [130] pour les données textuelles, les descripteurs SIFT [96] et HoG [60] pour les images, la courbure et les images de spin [86] pour les formes 3D, les descripteurs en ondelettes [98] pour le traitement du signal, et, plus généralement, le résultat d'une technique de réduction de dimension, comme l'ACP, MDS ou Isomap [132]. L'efficacité des descripteurs est souvent corrélée aux propriétés dont ils bénéficient. En fonction de l'application, il peut être pertinent d'exiger d'un descripteur qu'il soit invariant par translation ou rotation, intrinsèque ou extrinsèque, un vecteur Euclidien, etc. Trouver des descripteurs avec de telles propriétés est une question importante car permettant d'améliorer grandement l'interprétation et la visualisation des données, comme mentionné plus haut, mais aussi le résultat des algorithmes d'apprentissage, qui sont susceptibles de produire de mauvaises performances si alimentés avec des données brutes. Le but de cette thèse est d'étudier une classe spécifique de descripteurs appelés *topologiques*, et qui sont connus pour être invariants aux déformations continues des données qui n'impliquent pas de déchirement ou de recollement [26].

1.1.2 Descripteurs topologiques

L'idée derrière les descripteurs topologiques est de synthétiser *l'information topologique* présente dans les données [26]. Intuitivement, la topologie des données englobe toutes les propriétés qui sont préservées par des déformations continues, comme l'étirement, le rétrécissement ou l'épaississement, sans déchirure ni recollement. Par exemple, si un cercle est continument déformé sans déchirement ou recollement, un trou va toujours subsister dans l'objet résultant, quelle qu'ait été la transformation. C'est ce qu'on appelle un *attribut topologique*. Voir la Figure 1.1, où la présence d'un trou est attestée dans différentes déformations du cercle.

De manière similaire, les composantes connexes, cavités, et trous de dimension supérieure sont des attributs topologiques. Dans l'optique de formaliser la présence de tels attributs (en toute dimension), *la théorie de l'homologie*, a été développée au 19e et au début du 20e siècle. Elle se présente comme un encodage algébrique de l'information topologique. L'homologie d'un espace est une famille de groupes abéliens (un pour chaque dimension), dont les éléments sont des combinaisons linéaires des trous de l'espace.

Cependant, les groupes d'homologie ne sont pas des descripteurs topologiques très performants en tant que tels, la raison principale étant que les données prennent souvent la forme de nuages de points, dont les groupes d'homologie ne sont pas informatifs : chaque point du nuage est un générateur du groupe d'homologie en dimension 0, puisque l'homologie en dimension 0 compte les composantes connexes, et tous les groupes d'homologie de dimension supérieure sont triviaux puisque le nuage n'a aucun trou. Evidemment, le nuage de points peut tout de même refléter de l'information topologique - par exemple s'il est échantillonné sur un objet géométrique comme un cercle, une sphère ou un tore. La question devient ainsi celle de l'échelle avec laquelle observer les données, comme illustré dans la Figure 1.2.

L'analyse de données topologiques fournit deux constructions : les diagramme de persistance, qui synthétisent l'information topologique à toutes les échelles, et les Mappers, qui encodent plus d'information

géométrique à échelle fixée.

Diagrammes de persistance. Puisque chaque échelle fournit des informations topologiques pertinentes, l'idée de l'homologie persistante est d'encoder l'homologie du nuage de points à toutes les échelles. Considérons la base de données de la Figure 1.3, contenant des images à 128 × 128 pixels, vus comme des vecteurs en dimension 16 384, où chaque coordonnée est le niveau de gris d'un pixel. Puisque la caméra a tourné autour de l'objet, il s'ensuit qu'à petite échelle, les données semblent être réparties en petits groupes, tandis qu'à échelle plus grande, elles semblent échantillonnées sur un cercle (plongé dans $\mathbb{R}^{16384}$).

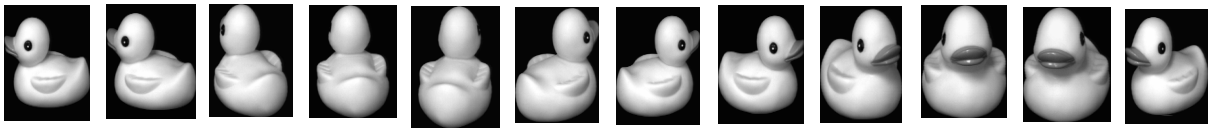

Figure 1.3: Une base de données d'images.

Pour synthétiser cette information, on peut faire grossir des boules centrées sur les points de données. Considérons trois rayons différents pour ces boules : un petit α, un légèrement plus grand β et un beaucoup plus grand γ, comme montré dans la Figure 1.4.

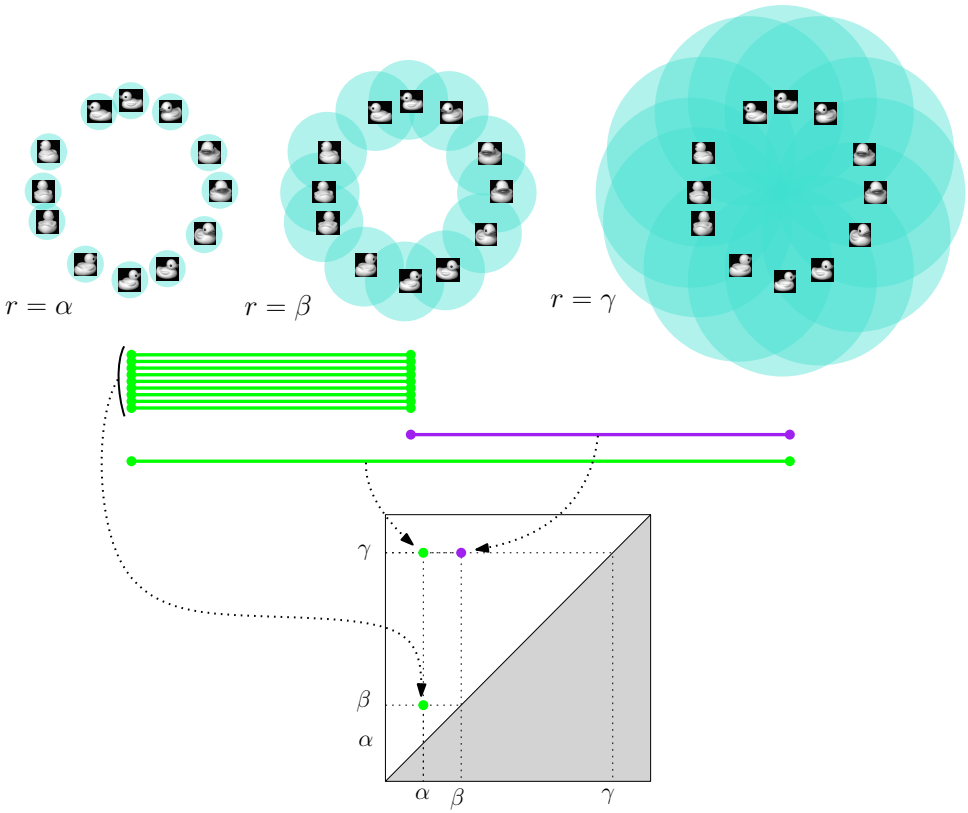

Figure 1.4: Trois différentes unions de boules centrées sur des images vus comme des vecteurs dans un espace Euclidien de grande dimension. L'apparition et la disparition d'attributs topologiques, comme des composantes connexes ou des trous, est enregistrée dans un *diagramme de persistance*, dans lequel les points représentant des attributs en dimension 0 sont en vert, et ceux représentant des attributs en dimension 1 sont en violet.

Quand le rayon des boules vaut α, l'union des boules est simplement l'union de dix composantes connexes, dont l'homologie en dimension 1 et supérieure est triviale. Cependant, quand le rayon devient

β, l'union des boules a l'homologie d'un cercle, dont le trou en dimension 1 devient rempli quand le rayon devient γ. On dit que les composantes connexes sont nées à la valeur α, et neuf sont mortes, c'est-à-dire se sont fait relier à la dixième, à la valeur β. De la même manière, le trou en dimension 1 est apparu au rayon β, et a disparu au rayon γ. Enfin, la dixième composante connexe est apparue au rayon α et a persisté jusqu'au rayon γ. Cette information est encodée dans le *diagramme de persistance*, qui est un multi-ensemble [1] de points, chacun représentant un attribut topologique, et ayant les rayons de naissance et de mort comme coordonnées. La distance à la diagonale fournit une quantité utile et interprétable dans les diagrammes de persistance. En effet, si un point est loin de la diagonale, alors son ordonnée est largement supérieur à son abscisse, ce qui signifie que l'attribut topologique correspondant était présent dans l'union des boules pour une large gamme de rayons différents, indiquant ainsi que l'attribut topologique a des chances d'être présent dans l'objet sous-jacent, et d'être une information pertinente. Au contraire, les points proches de la diagonale représentent des attributs qui ont disparu rapidement après être apparus. Ces attributs éphemères correspondent plutôt à du bruit ou des attributs de l'objet sous-jacent qui ne sont pas pertinents. C'est le cas par exemple des neuf composantes connexes de l'union des boules au rayon α dans la Figure 1.4, qui ont disparu au rayon β, proche de α. Il est à noter que nous avons éxpliqué la construction dans le cas où il n'y a que trois unions de boules, mais il est bien sûr possible de construire un diagramme de persistance quand le rayon des boules augmente continument de 0 à $+\infty$. Dans ce cas, le trou de dimension 1 a une abscisse située entre α et β (car il n'est pas encore présent pour le rayon α et est déjà là au rayon β), et une ordonnée située entre β et γ (car il a déjà disparu au rayon γ). De même, toutes les composantes connexes ont pour abscisse 0. Neuf d'entre elles[2] ont une ordonnée comprise entre α et β et l'ordonnée de la dixième est $+\infty$ puisqu'elle est toujours présente, quelque soit le rayon des boules.

Les diagrammes de persistance peuvent en faire être définis beaucoup plus généralement. - même si l'interprétation en terme d'échelle n'est plus forcément pertinente. Tout ce qui est requis est une famille d'espaces intriqués les uns dans les autres, appelée *filtration*, c'est-à-dire une famille $\{X_\alpha\}_{\alpha\in A}$, où A est un ensemble d'indices totalement ordonnés, telle que $\alpha \leq \beta \Rightarrow X_\alpha \subseteq X_\beta$. La construction du diagramme de persistance est alors la même, c'est-à-dire l'enregistrement de l'apparition et de la disparition d'attributs topologiques quand on parcourt A par ordre croissant. Dans l'exemple précédent, la filtration contient trois espaces, qui sont les trois différentes unions de boules, chaque union étant indicée par le rayon de ses boules. Il est clair dans ce cas que ces trois espaces sont intriqués car une boule est toujours incluse dans la boule de même centre avec un rayon supérieur.

Une manière pratique de construire une filtration est d'utiliser les *sous-niveaux* d'une fonction continue à valeurs réelles f, c'est-à-dire les espaces de la forme $f^{-1}((-\infty,\alpha])$. En effet, il est évident que $f^{-1}((-\infty,\alpha]) \subseteq f^{-1}((-\infty,\beta])$ pour tous $\alpha \leq \beta \in \mathbb{R}$. Par exemple, l'union des boules de rayon r centrées sur les points d'un nuage P est égale au sous-niveau de la fontion distance au nuage P : $d_P^{-1}((-\infty,r])$, où $d_P(x) = \min_{p\in P} d(x,p)$. Ainsi, dès qu'une fonction continue à valeurs réelles est à disposition, un diagramme de persistance peut être construit, ce qui explique pourquoi le diagramme de persistance est un descripteur prolifique. Prenons par exemple l'image floue d'un zéro, affichée dans le coin inférieur droit de la Figure 1.5, pour laquelle le niveau de gris des pixels est utilisé comme fonction continue pour calculer un diagramme de persistance. De nouveau, on trouve deux points se distinguant des autres dans le diagramme de persistance, l'un représentant la composant connexe du zéro, et l'autre son trou de dimension 1. Le reste des points est engendré par le bruit présent dans l'image.

Une des raisons pour lesquelles les diagrammes de persistance sont des descripteurs appréciés est qu'en plus d'être invariant par déformation continue (sans déchirement ou recollement), ils sont *stables* [42, 54]. En effet, si des diagrammes de persistance sont calculés avec les sous-niveaux de fonctions similaires, alors la distance entre eux est bornée supérieurement par la différence entre les fonctions en norme infinie :

$$d_{\mathrm{b}}(\mathrm{Dg}(f),\mathrm{Dg}(g)) \leq \|f-g\|_\infty,$$

où d_{b} désigne la distance bottleneck entre diagrammes de persistance, qui est le coût de la meilleure correspondance partielle entre les points de chaque diagramme. Cela signifie que, par exemple, si les positions des images de la Figure 1.4 sont légèrement perturbées, ou si l'image floue du zéro de la Figure 1.5 est légèrement modifiée, les diagrammes de persistance correspondant seront très proches des

[1] Un multi-ensemble est une généralisation d'un ensemble, dans laquelle les points ont des multiplicités.

[2] En fait, chaque point est une composante connexe au rayon 0.

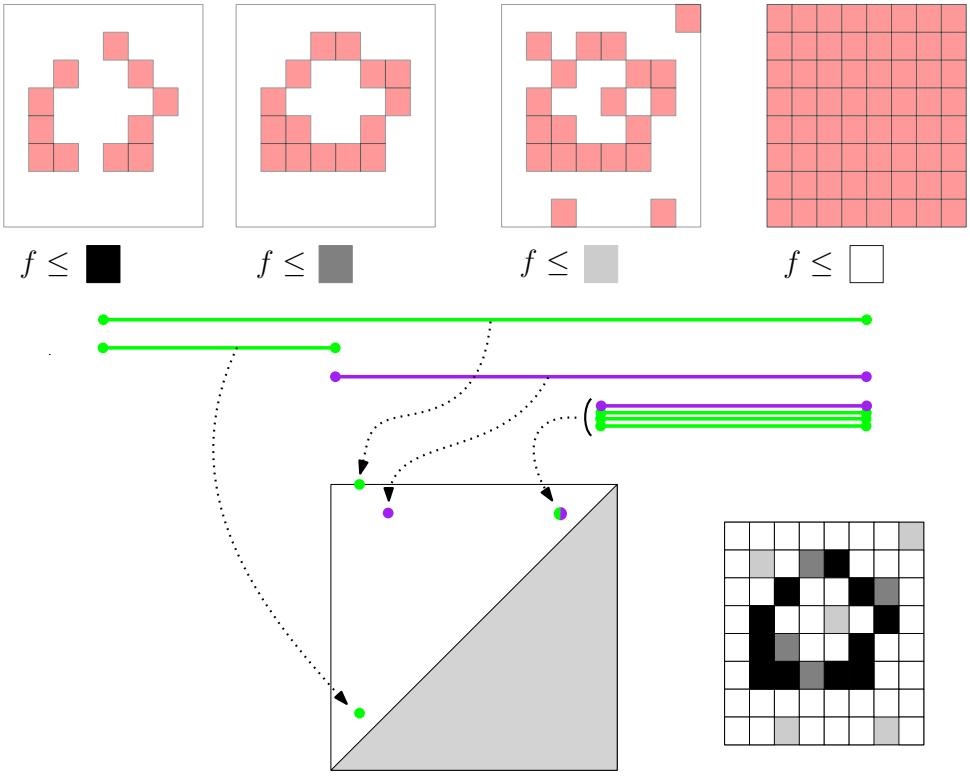

Figure 1.5: Autre exemple d'une construction de diagramme de persistance, avec les sous-niveaux du niveau de gris des pixels d'une image floue d'un zéro.

originaux avec la distance bottleneck.

Les diagrammes de persistance ont aidé à améliorer l'analyse des données dans de nombreuses applications, allant de l'analyse de forme 3D [38, 43] à la transition de phase de matériaux [73, 84] et la génomique [24, 39] pour n'en citer que quelques-unes.

Mapper. Comme expliqué plus haut, les diagrammes de persistance synthétisent l'information de nature topologique contenue dans les données. Cependant, ils perdent beaucoup d'information géométrique dans le processus : ils est aisé de construire des espaces différents ayant les mêmes diagrammes de persistance. Le *Mapper*[3], introduit par [129], est une approximation directe de l'objet sous-jacent, qui contient non seulement les attributs topologiques, mais aussi de l'information additionnelle, concernant le positionnement des attributs les uns par rapport aux autres par exemple. Comme pour les diagrammes de persistance, une fonction réelle continue, appelée parfois *filtre*, est requise, ainsi qu'une couverture de son image par des intervalles ouverts qui se chevauchent. L'idée est de calculer les antécédents par f de tous les intervalles de la couverture, de les raffiner en leurs composantes connexes via des techniques de clustering, et de finalement lier les composantes connexes entre elles si elles contiennent des points de données en commun.

Nous fournissons un exemple dans la Figure 1.6, où nous considérons de nouveau le nuage d'images. La fonction réelle continue est la valeur absolue de l'angle à partir duquel l'image a été prise, et son image $[0, \pi]$ est couverte par trois intervalles (bleu, rouge et vert). Dans les antécédents des intervalles rouge et bleu, il y a une seule composant connexe, tandis qu'il y en a deux dans l'antécédent de l'intervalle vert. Le Mapper est obtenu en ajoutant des arètes entre les composantes connexes, en fonction de la présence ou non de points de données en commun à l'intérieur de ces composantes; par exemple, les composantes connexes vertes et bleues, ou vertes et rouges, sont reliées, mais pas celles qui sont rouges et bleues. Le Mapper a l'homologie d'un cercle, est constitue une approximation directe du support sous-jacent au

[3]Dans cette thèse, on appelle *Mapper* l'objet mathématique, et pas l'algorithme utilisé pour le construire.

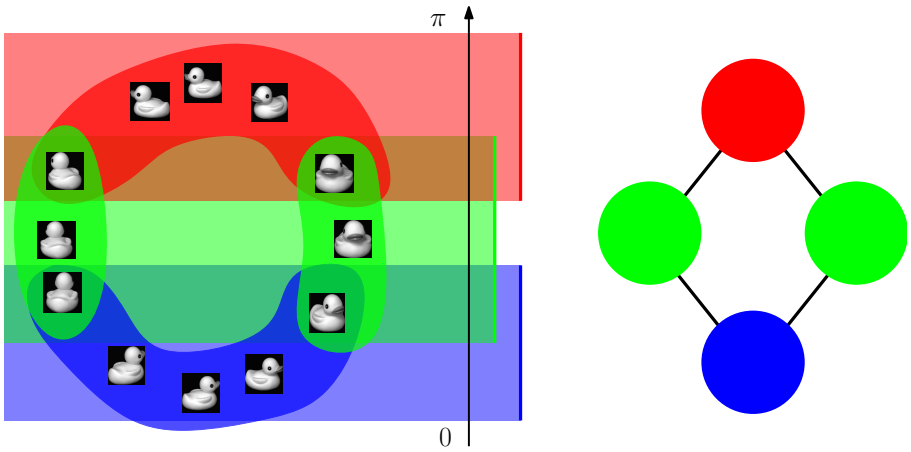

Figure 1.6: Exemple de Mapper calculé sur le nuage d'images, avec la fonction d'angle et une couverture de trois intervalles.

nuage d'images.

Il est bon de remarquer que les longueurs des intervalles contrôlent directement l'échelle à partir de laquelle on observe le nuage : si les intervalles sont petits, le Mapper va avoir beaucoup de composantes déconnectées puisque les antécédents contiendront au plus un point de donnée. A l'opposé, si les intervalles sont larges, le Mapper aura peu de composantes puisque les antécédents vont contenir beaucoup de points de données.

En pratique, le Mapper a deux domaines d'applications majeures. Le premier est la visualisation et le clustering. En effet, le Mapper fournit une visualisation des données sous forme de graphe dont la topologie reflète celle des données. Il apporte ainsi une information complémentaire à celle des algorithmes de clustering usuels concernant la structure interne des clusters par l'identification de *branches* et de *boucles* qui mettent en lumière des attributs topologiques potentiellement remarquables dans les groupes identifiés par clustering. Voir par exemple [138, 97, 125, 83] pour des exemples d'applications. La deuxième application est la sélection d'attributs. En effet, chaque attribut des données peut être évalué en regard de sa capacité à différencier les attributs topologiques mentionnés plus haut (branches et boucles) du reste des données, via l'utilisation de tests statistiques, comme celui de Kolmogorov-Smirnov. Voir par exemple [97, 109, 122] pour des exemples d'applications.

1.1.3 Principales limitations

Même si le Mapper et les diagrammes de persistance bénéficient de propriétés désirables, plusieurs limitations refrènent leur usage pratique, à savoir la *la difficulté de la sélection de paramètres pour Mapper* et la *non linéarité* de l'espace des diagrammes de persistance.

Distance et stabilité pour les Mappers et les graphes de Reeb Un problème du Mapper est que, contrairement aux diagrammes de persistance, il a un paramètre, la couverture, dont la sélection à priori est difficile. A cause de cela, le Mapper apparaît comme une construction très *instable* : il arrive que des Mappers calculés sur des nuages de points similaires, comme dans la Figure 1.7, ou avec des couvertures proches, comme dans la Figure 1.8, soient très différents.

Ce problème majeur est un obstacle important à son utilisation en exploration de données. La seule réponse dans l'état-de-l'art consiste à sélectionner des paramètres dans une grille de valeurs pour lesquels le Mapper semble stable - voir [109] par exemple.

Ainsi, prouver un résultat de stabilité pour les Mappers nécessite de les comparer avec une distance qui dépend au moins de la couverture utilisée. Malheureusement, même si des distances théoriques peuvent être définies [105], la définition d'une distance calculable et interprétable entre Mappers manque dans l'état-de-l'art. Pour gérer ce problème, on peut prendre inspiration d'une classe de descripteurs très semblables aux Mappers, les *graphes de Reeb*.

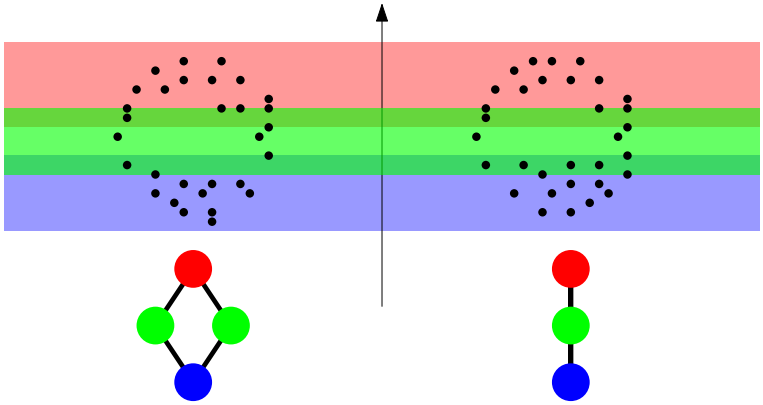

Figure 1.7: Mappers calculés sur des échantillonnages similaires du cercle, avec la fonction hauteur et une couverture composée de trois intervalles.

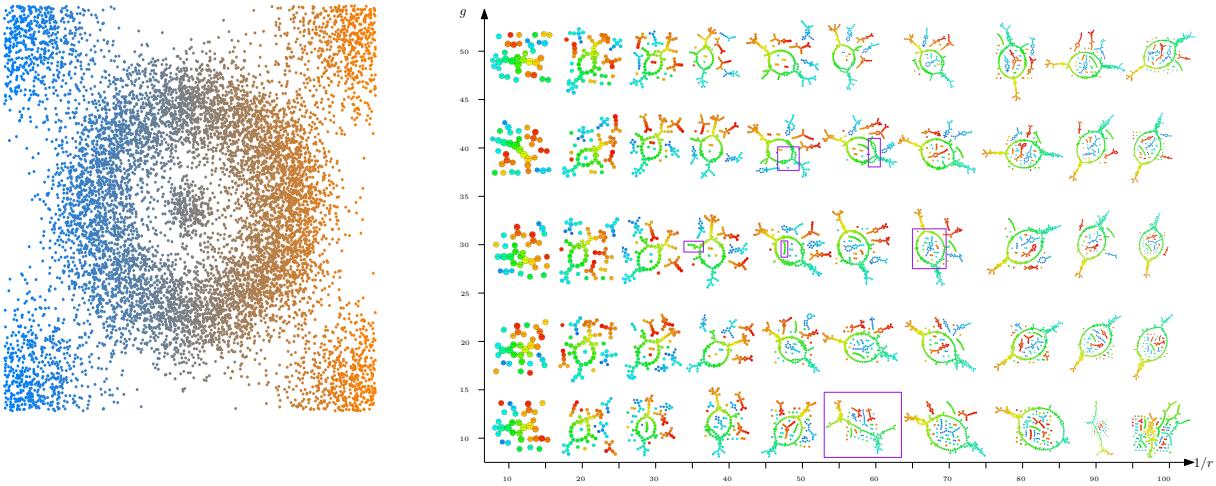

Figure 1.8: Un ensemble de Mappers calculés sur le jeu de données du cratère avec des couvertures différentes (r est la longueur des intervalles et g est le pourcentage de chevauchement) et la coordonnée horizontale. Gauche : jeu de données du cratère coloré par les valeurs de fonction, allant de bleu à orange. Droite : Mappers calculés avec des paramètres différents. Les rectangles violets indiquent les attributs topologiques qui apparaissent ou disparaissent soudainement dans les Mappers.

Graphes de Reeb. Même si les Mappers sont définis pour des nuages de points, leur extension à des espaces non discrets est évidente, la différence étant que des techniques de clustering ne sont pas nécessaires pour calculer les composantes connexes des antécédents puisqu'elles sont bien définies. Dans ce cas, faire tendre la longueur des intervalles vers zéro définit le *graphe de Reeb*. Ainsi, les Mappers (calculés sur des espaces non discrets) ne sont que des *approximations*, ou des *versions pixelisées* des graphes de Reeb, comme illustré dans la Figure 1.9.

Cette observation est cruciale car plusieurs distances, ainsi que des résultats de stabilité, ont été obtenus pour les graphes de Reeb [7, 8, 61] et peuvent être étendus aux Mappers. Cependant, ces distances ne sont pas calculables et ne peuvent pas être utilisées en tant que telles en pratique [2]. La question de savoir s'il est possible de définir des distances stables et calculables pour les Mappers reste ainsi ouverte.

Non linéarité de l'espace des diagrammes de persistance. Même si les diagrammes de persistance sont stables, ils ne peuvent pas être utilisés systématiquement par des algorithmes d'apprentissage automatique. En effet, une classe très large de ces algorithmes nécessitent que les données soient soit des vecteurs d'un espace Euclidien (comme les forêts aléatoires), ou d'un espace de Hilbert (comme les SVM).

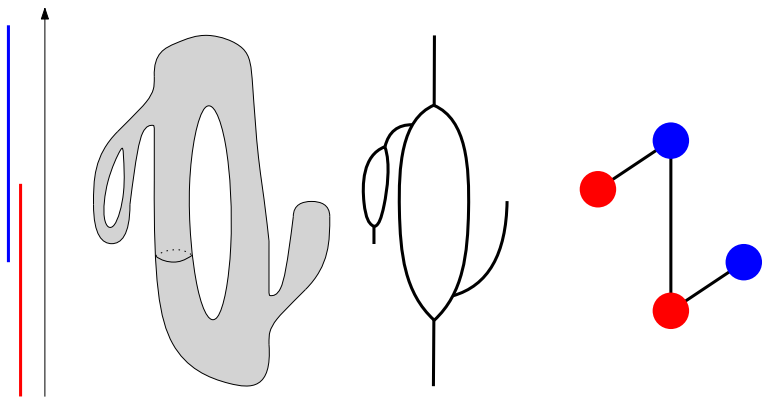

Figure 1.9: Une surface plongée dans $\mathbb{R}^3$ (gauche), son graphe de Reeb calculé avec la fonction hauteur (milieu) et son Mapper calculé avec la fonction hauteur et une couverture à deux intervalles (droite).

L'espace des diagrammes de persistance, équipé avec la distance bottleneck, n'est malheureusement ni l'un ni l'autre. Même les moyennes de Fréchet ne sont pas bien définies [136]. L'*astuce du noyau* permet cependant de traiter ce genre de données. En supposant que les points de données vivent dans un espace métrique (X, d_X), l'astuce du noyau nécessite seulement une fonction semi-définie positive, appelée *noyau*, c'est-à-dire une fonction $k : X \times X \to \mathbb{R}$ telle que, pour tous $a_1, \cdots, a_n \in \mathbb{R}$ et $x_1, \cdots, x_n \in X$, on ait :

$$\sum_{i,j} a_i a_j k(x_i, x_j) \geq 0.$$

Grâce au théorême de Moore-Aronszajn [4], les valeurs du noyau calculées sur des points de données peuvent être démontrées égales à l'évaluation d'un produit scalaire entre les images des points de données par un plongement dans un espace de Hilbert spécifique qui dépend uniquement de k et qui est en général inconnu. Plus formellement, il existe un espace de Hilbert $\mathcal{H}_k$ tel que, pour tous $x, y \in X$, on ait :

$$k(x, y) = \langle \Phi_k(x), \Phi_k(y) \rangle_{\mathcal{H}_k},$$

pour un certain plongement Φ_k. Les valeurs du noyau peuvent donc être considérées comme des produits scalaires généralisés entre les points de données, et peuvent être directement utilisés par les algorithmes d'apprentissage. Dans le cas qui nous intéresse, la question est ainsi de trouver de tels noyaux pour les diagrammes de persistance.

Une manière standard de procéder pour définir un noyau pour des points d'un espace métrique (X, d_X) est d'utiliser des fonctions *Gaussiennes* :

$$k_\sigma(x, y) = \exp\left(-\frac{d_X(x, y)}{2\sigma^2}\right),$$

où $\sigma > 0$ est un paramètre d'échelle. Un théorême de Berg et al. [11] stipule que k_σ est un noyau, c'est-à-dire une fonction semi-définie positive, pour tous $\sigma > 0$ si et seulement si d_X est *conditionnellement semi-définite négative*, c'est-à-dire est telle qu'on ait $\sum_{i,j} a_i a_j d_X(x_i, x_j) \leq 0$ pour tous $x_1, \cdots, x_n \in X$ et $a_1, \cdots, a_n \in \mathbb{R}$ tels que $\sum_{i=1}^n a_i = 0$. Malheureusement, comme montré par Reininghaus et al. [119], la distance bottleneck d_b pour les diagrammes de persistance n'est pas conditionnellement semi-définite négative. Il est même possible de trouver des contre-exemples pour les distances de Wasserstein, une autre classe de distance pour diagrammes. L'utilisation de noyaux Gaussiens pour les diagrammes de persistance est donc impossible avec leurs métriques canoniques.

Néanmoins, plusieurs noyaux ont été proposés au cours des dernières années [1, 20, 90, 120], bénéficiant tous de résultats de stabilité bornant supérieurement la distance entre les plongements des diagrammes par les distances bottleneck ou de Wasserstein entre les diagrammes eux-mêmes. En d'autres termes, la distorsion métrique

$$\mathrm{dist}(\mathrm{Dg}, \mathrm{Dg}') = \frac{\|\Phi_k(\mathrm{Dg}) - \Phi_k(\mathrm{Dg}')\|_{\mathcal{H}_k}}{d_\mathrm{b}(\mathrm{Dg}, \mathrm{Dg}')}$$

est bornée supérieurement. Cependant, le calcul d'une borne inférieure non triviale reste ouvert : il se pourrait que les plongements de diagrammes différents soient en fait très proches l'un de l'autre, ce qui n'est pas désirable en pratique pour la discriminativité d'un noyau. Par exemple, le plongement constant, qui envoie tous les diagrammes sur un même point d'un espace de Hilbert spécifique, est stable (les distances entre images dans l'espace de Hilbertétant toujours nulles), mais les résultats du noyau correspondant seront évidemment très faibles. Plus généralement, le comportement et les propriétés des distances dans les espaces de Hilbert induits par des noyaux sont flous, et la question de savoir s'il existe des noyaux avec des propriétés théoriques de discriminativité est ouverte.

1.1.4 Contributions

Dans cette thèse, nous nous penchons sur trois problèmes : l'interprétation des attributs topologiques) du Mapper (par exemple avec des régions de confiance), le réglage de ses paramètres, et l'intégration globale des descripteurs topologiques en apprentissage automatique.

Distance entre graphes de Reeb. Dans le Chapitre 3, nous définissons une pseudodistance calculable entre graphes de Reeb, qui revient à comparer leurs diagrammes de persistance. Nous montrons aussi que cette pseudodistance est en fait *localement équivalente* aux autres distances existantes pour les graphes de Reeb. Cette équivalence locale est alors utilisée pour étudier les propriétés de l'espace métrique des graphes de Reeb, équipé des distances *intrinsèques*. Nous montrons que toutes ces distances intrinsèques sont *fortement équivalentes*, ce qui nous permet d'englober toutes les techniques pour comparer des graphes de Reeb en une seule approche. Ce travail a été publié dans les proceedings du Symposium on Computational Geometry 2017 [36].

Structure du Mapper. Dans le Chapitre 4, nous fournissons un lien entre les diagrammes de persistance du graphe de Reeb et ceux du Mapper (calculé sur le même espace topologique). Plus spécifiquement, nous montrons que le diagramme de persistance du Mapper est obtenu à partir de celui du graphe de Reeb en supprimant des points spécifiques, à savoir ceux qui appartiennent à des régions du plan qui dépendent uniquement de la couverture utilisée pour calculer le Mapper. Cette relation explicite nous permet alors d'étendre la pseudodistance entre graphes de Reeb aux Mappers. Nous montrons finalement que cette pseudodistance *stabilise* les Mappers : nous fournissons un théorême de stabilité pour des Mappers comparés avec cette pseudodistance. Ce travail a été publié dans les proceedings du Symposium on Computational Geometry 2016 [35] et une version longue a été soumise au Journal of Foundations of Computational Mathematics [34].

Cas discret. Dans le Chapitre 5, nous étendons les résultats précédents au cas où les Mappers sont calculés sur des espaces discrets, c'est-à-dire des nuages de points, et les composantes connexes sont calculées avec du single-linkage clustering. En particulier, nous fournissons des conditions suffisantes pour lesquelles le Mapper calculé sur un nuage de points coincide avec celui calculé sur le support. De plus, nous montrons que le Mapper converge vers le graphe de Reeb avec une vitesse de convergence *optimale*, au sens où aucun estimateur du graphe de Reeb ne peut converger plus vite. Les paramètres utilisés pour démontrer l'optimalité fournissent en plus des *heuristiques pour le réglage automatique* de ces paramètres. Ces heuristiques se basent sur des techniques de sous-échantillonnage et dépendent uniquement de la cardinalité du nuage de points de données. Finalement, nous proposons un moyen de calculer des *régions de confiance* pour les différents attributs topologiques du Mapper. Ce travail a été soumis au Journal of Machine Learning Research [33].

Méthodes à noyaux. Dans le Chapitre 6, nous appliquons des techniques d'apprentissage aux diagrammes de persistance, via des méthodes à noyaux.

Nous définissons d'abord un *noyau Gaussien* en utilisant une modification de la distance de Wasserstein, appelée distance de *Sliced Wasserstein*. Nous montrons en effet que cette distance, à l'inverse de la distance de Wasserstein, est bien conditionnellement semi-définie négative, et permet donc de définir un noyau Gaussien. De plus, nous montrons que la distance induite dans l'espace de Hilbert associé est *équivalente* à la distance de Wasserstein de départ. Ainsi, ce noyau, en plus d'être stable et Gaussien, est aussi théoriquement discriminant. Nous en fournissons aussi une preuve empirique en obtenant de nettes

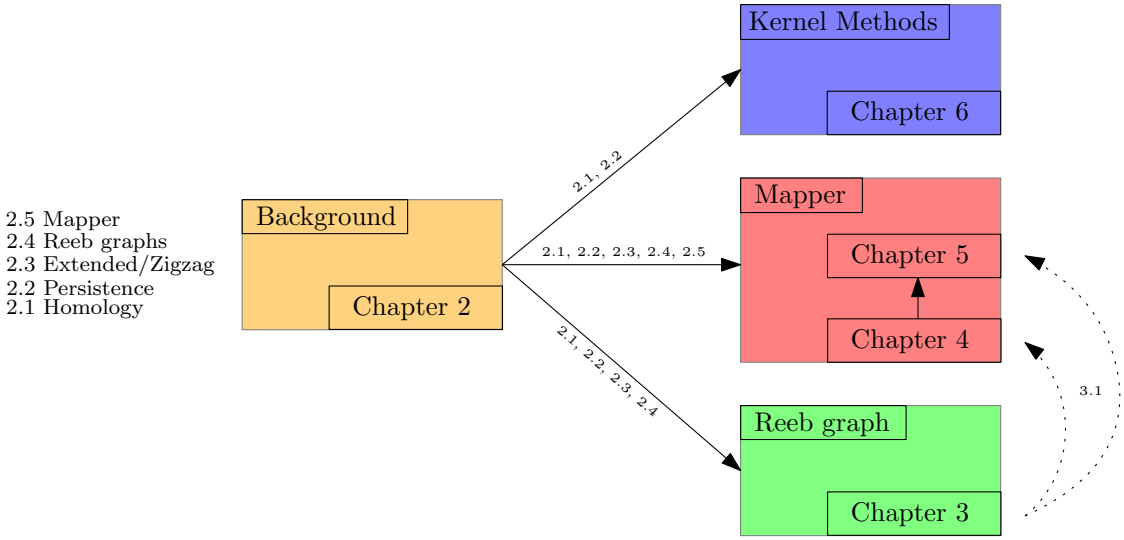

Figure 1.10: Les flèches indiquent des dépendances entre chapitres, et les flèches en pointillés indiquent des dépendances partielles, c'est-à-dire que seule une petite et non essentielle partie du chapitre dépend de l'autre.

améliorations par rapport aux autres noyaux de l'état-de-l'art dans plusieurs applications. Ce travail a été publié dans les proceedings de l'International Conference on Machine Learning 2017 [32].

Enfin, nous définissons aussi une *méthode de vectorisation* pour envoyer les diagrammes de persistance dans $\mathbb{R}^D$, où $D \in \mathbb{N}^*$. Ce plongement stable, même si non injectif, permet l'usage des diagrammes de persistance pour des problèmes et algorithmes où des vecteurs Euclidiens sont nécessaires. Nous détaillons alors une application où une telle structure est requise, à savoir le traitement de formes 3D, pour laquelle nous démontrons que les diagrammes de persistance apportent une information complémentaire aux descripteurs traditionnels. Ce travail a été publié dans les proceedings du Symposium on Geometry Processing 2015 [38].

Comment lire cette thèse ? Cette thèse est composée de quatre parties différentes :

- La première est le Chapitre 2, dans lequel nous détaillons les fondations théoriques de l'homologie, la persistance, les graphes de Reeb et les Mappers. Nous expliquons aussi la *persistance étendue* et la *persistance en zigzag*.
- La deuxième partie est le Chapitre 3, qui traite des graphes de Reeb et de leurs distances.
- La troisième partie est composée des Chapitres 4 et 5, qui traitent de Mapper.
- La quatrième partie est le Chapitre 6. Il traite des noyaux pour les diagrammes de persistance, dans des espaces de Hilbert en dimension finie et infinie.

Voir la Figure 1.10. Le Chapitre 2 rappelle essentiellement les fondamentaux en topologie. Les autres chapitres contiennent en revanche les contributions de cette thèse. Les Chapitres 3 et 4 sont très orientés topologie, tandis que le Chapitre 5 utilise plutôt des notions de statistiques, et que le Chapitre 6 se concentre davantage sur l'apprentissage automatique. Ces chapitres ne sont pas indépendants, comme illustré par la Figure 1.10, mais les contributions de chaque chapitre sont énoncées dans les introductions correspondantes. Ainsi, pour chacun de ces chapitres, le lecteur, en fonction de ses goûts ou connaissances personnelles, peut soit se limiter à l'introduction, soit lire le chapitre dans son intégralité.

1.2 Introduction in english

1.2.1 Data Analysis and Machine Learning

Data collection and generation in various human activities, including both industry and academia, have grown exponentially over the last decade and are now ubiquitous in many different fields of science, finance and industry. For example, in digital science, the fast development of image acquisition and processing has allowed large amounts of images to become publicly available online [93, 89, 107, 112, 123]. Similarly, in biology, next generation high-throughput sequencing allowed most laboratories to easily determine DNA sequences of sample organisms [14, 78, 88, 100]. Hence, the need to summarize and extract useful information from these massive amounts of data has become a problem of primary interest.

Machine Learning is a field of data science that aims at deriving algorithms ("Machine") that can make predictions about new data solely from the information that is contained in already collected datasets ("Learning"). These techniques can provide answers to multiple data analysis problems such as *classification*, which aims at predicting labels, *clustering*, which aims at separating data into groups or clusters, or *regression*, which aims at approximating functions on data. We refer the interested reader to [72] for a comprehensive introduction to these methods. For example, a typical classification problem would be to predict if a drug have effects on a specific patient P. This is a binary classification problem since the label we want to predict for P is either "effect" or "no effects". Assuming you have a database of patients at hand, in which the drug effects on each patient were recorded, one of the simplest way to proceed is to look for the closest match, or most similar patient, to P in this database, and to take the label of this match. This extremely simple yet powerful method is called *nearest neighbor* prediction, and has been extensively studied by data scientists. More generally, nearest neighbor prediction is nothing but a small part of the large variety of methods proposed in Machine Learning, which can tackle many real-life challenges including image classification, musical genre prediction or medical prognosis to name a few. More examples of applications and datasets can be found in [72].

Descriptors. Usually, data comes in the form of a point cloud in $\mathbb{R}^D$, where $D \in \mathbb{N}^*$. Each data point represents an *observation* and each dimension, or coordinate, represents a *measurement*. For instance, observations can be patients, images or DNA sequences, whose corresponding measurements are physical characteristics (height, weight, age...), grey scale values of pixels or nucleobases A,C,T,G composing DNA. It is often the case that the number of measurements is very large, leading to a rich level of information, but also making data very high-dimensional and impossible to visualize.

Hence, a large part of data analysis is devoted to the summarization of the information contained in datasets or data points into simple and interpretable *descriptors* or *signatures*, which are usually application specific. For instance, among common descriptors are: bag-of-words models [130] for text document data, SIFT [96] and HoG [60] descriptors for image data, curvature and spin images [86] for 3D shape data, wavelet descriptors [98] for signal data, and, more generally, outputs of data reduction technique, such as MDS, PCA or Isomap [132]. The efficiency of descriptors is very often correlated to the properties they enjoy. Depending on the application, one may want descriptors to be translation or rotation invariant, intrinsic or extrinsic, to lie in Euclidean space, etc. Deriving descriptors with desirable properties is important since it greatly enhances interpretation and visualization, as mentioned above, but it also improves the performances of Machine Learning algorithms, which may perform poorly if fed with raw data. The aim of this thesis is to study a specific class of descriptors called *topological* descriptors, which are known to be invariant to continuous deformations of data that do not involve tearing or gluing [26].

1.2.2 Topological Descriptors

The idea of topological descriptors is to summarize the *topological information* contained in data [26]. Intuitively, the topology of data encompasses all of its properties that are preserved under continuous deformations, such as stretching, shrinking or thickening, without tearing or gluing. For instance, when a circle is continuously deformed without tearing or gluing, the hole always remains in the resulting object, whatever the transformation. This is a topological *invariant* or *feature*. See Figure 1.11, where a hole is always present in the displayed deformations of the circle.

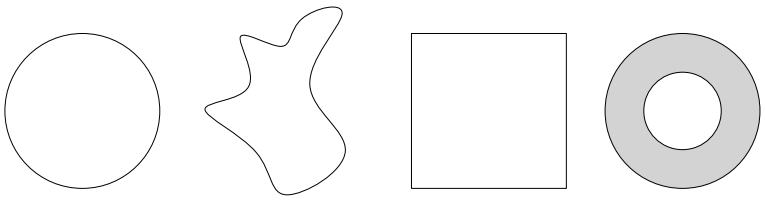

Figure 1.11: Deformations of a circle.

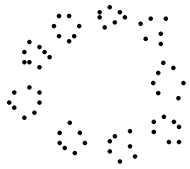

Figure 1.12: This point cloud seems to be sampled on nine circles from a small scale, and on a single circle from a larger scale.

Similarly, connected components, cavities and higher-dimensional holes are topological features. In order to formalize the presence of such holes (in any dimension), *homology theory* was developed in the 19th and the beginning of the 20th century. It provides an algebraic encoding of such topological information. Basically, the homology of a space is a family of abelian groups (one for each topological dimension), whose elements are linear combinations of the space's holes.

However, it turns out that the homology groups themselves perform poorly as topological descriptors. The main reason is that data often comes in the form of point clouds, and the homology groups are not informative for such objects: each point of the cloud is a generator of the 0-dimensional homology group, since 0-dimensional homology is concerned with connected components, and all higher-dimensional homology groups are trivial since the point cloud has no holes. However, it may happen that the data still contains topological information, for instance when the point cloud is a sampling of a geometric object such as a circle, a sphere or a torus. Hence, the question that arises is that of the scale at which one should look at the data, as illustrated in Figure 1.12.

Topological data analysis provides two constructions: *persistence diagrams*, which summarize the topological information at all possible scales, and *Mappers*, which encode extra geometric information but at a fixed scale.

Persistence diagrams. Since several different scales may contain relevant topological information, the idea of persistent homology is to encode the homology of the point cloud at all possible scales. Consider the dataset of Figure 1.13, containing images with 128×128 pixels, seen as 16,384-dimensional vectors, where each coordinate is the grey scale value of a pixel. Since the camera circled around the object, it follows that, from a small scale, the data looks composed of small clusters, each of which characterizing a specific angle, whereas from a larger scale, the data seems to be sampled on a circle (embedded in $\mathbb{R}^{16,384}$).

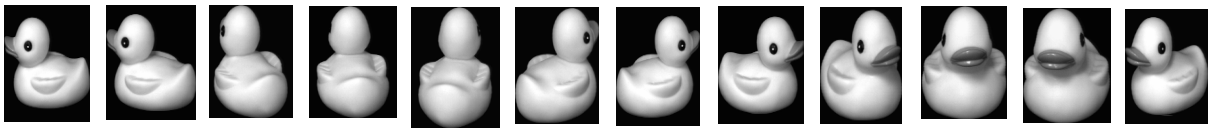

Figure 1.13: A dataset of images.

To summarize this information, the idea is to grow balls centered on each point of the dataset. Let us look at three different radius values: a small one α, a slightly larger intermediate one β, and a very large one γ for these balls, as displayed in Figure 1.14.

When the radius of the balls is α, the union of balls is a just a union of ten connected components with trivial homology in dimension 1 and above. However, when the radius is β, the union of balls has the homology of a circle, whose 1-dimensional hole gets filled in when the radius increases to γ. Hence, we say that the connected components were born at value α, and nine of them died, or got merged in the tenth one, at radius β. Similarly, the 1-dimensional circle was born, or appeared, at value β and died, or got filled in, at value γ. Finally, the tenth connected component appeared at radius α and remained all the way until radius γ. This is summarized in the *persistence diagram*, which is a multiset [4] of points, each of

[4] A multiset is a generalization of a set, in which points can have multiplicities.

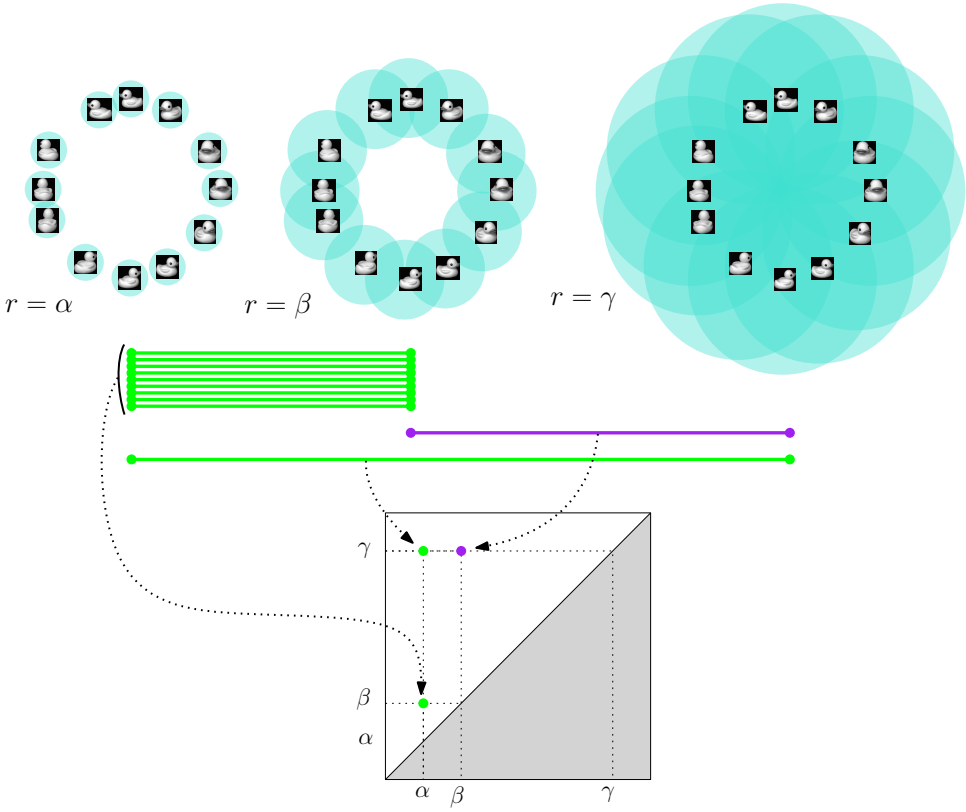

Figure 1.14: Three different unions of balls centered on images seen as vectors in high-dimensional Euclidean space. The appearance and disappearance of topological features like connected components or holes is recorded and stored in the so-called *persistence diagram*, in which green points represent 0-dimensional features and purple points represent 1-dimensional features.

which represents a topological feature and has the birth and death radii as coordinates. The distance to the diagonal is a useful interpretable quantity in persistence diagrams. Indeed, if a point is far from the diagonal, then its ordinate, or death radius, is much larger than its abscissae, or birth radius. This means that the corresponding topological feature was present in the union of balls for a large interval of radii, suggesting that the feature is likely to be present in the underlying object, and thus significant. On the contrary, points close to the diagonal represent features that disappeared quickly after their appearance. These fleeting features are likely to be nonsignificant features or noise artifacts. Consider for instance the nine connected components in the union of balls of radius α in Figure 1.14, which disappeared at radius β slightly larger than α. Note that we explained the construction using only three unions of balls, but it is of course possible to compute a persistence diagram when the radius increases continuously from 0 to $+\infty$. In that case, the 1-dimensional hole has an abscissa located between α and β (since it is not yet present at radius α and already present at radius β), and an ordinate located between β and γ (since it is already gone at radius γ). Similarly, all connected components have an abscissa equal to 0. Nine of them[5] have an ordinate located between α and β and the ordinate of the tenth one is $+\infty$ since it is always present, whatever the radius of the balls.

Persistence diagrams can actually be defined much more generally—even though the interpretation with scales may no longer be true. All that is needed is a family of spaces which is nested with respect to the inclusion, called a *filtration*. This is a family $\{X_\alpha\}_{\alpha \in A}$, where A is a totally ordered index set, such that $\alpha \leq \beta \Rightarrow X_\alpha \subseteq X_\beta$. Then, the construction of persistence diagrams remains the same, i.e. keeping track of the appearance and disappearance of topological features as we go through all indices in ascending order. In the previous example, the filtration had three elements, the three different unions of balls, each union being indexed by the radius of its balls. It is clear in this case that these three spaces

[5]Actually, each point is a connected component at radius 0.

are nested since a ball is always included in the ball with same center and larger radius.

A common way to build a filtration is to use the *sublevel sets* of a continuous scalar-valued function f, which are sets of the form $f^{-1}((-\infty, \alpha])$. Indeed, it is clear that $f^{-1}((-\infty, \alpha]) \subseteq f^{-1}((-\infty, \beta])$ for any $\alpha \leq \beta \in \mathbb{R}$. For instance, the union of balls with radius r centered on the points of a point cloud P is equal to the sublevel set of the distance function to P: $d_P^{-1}((-\infty, r])$, where $d_P(x) = \min_{p \in P} d(x, p)$. Hence, as soon as there is a continuous scalar-valued function at hand, a persistence diagram can be computed, which explains why the persistence diagram is a versatile descriptor. Consider for instance the blurry image of a zero in the down right corner of Figure 1.15, where the grey value function is used to compute the persistence diagram. Again, there are two points standing out in the persistence diagram, one representing the connected component of the zero, and the other representing the 1-dimensional hole induced by the zero. All other points are noise.

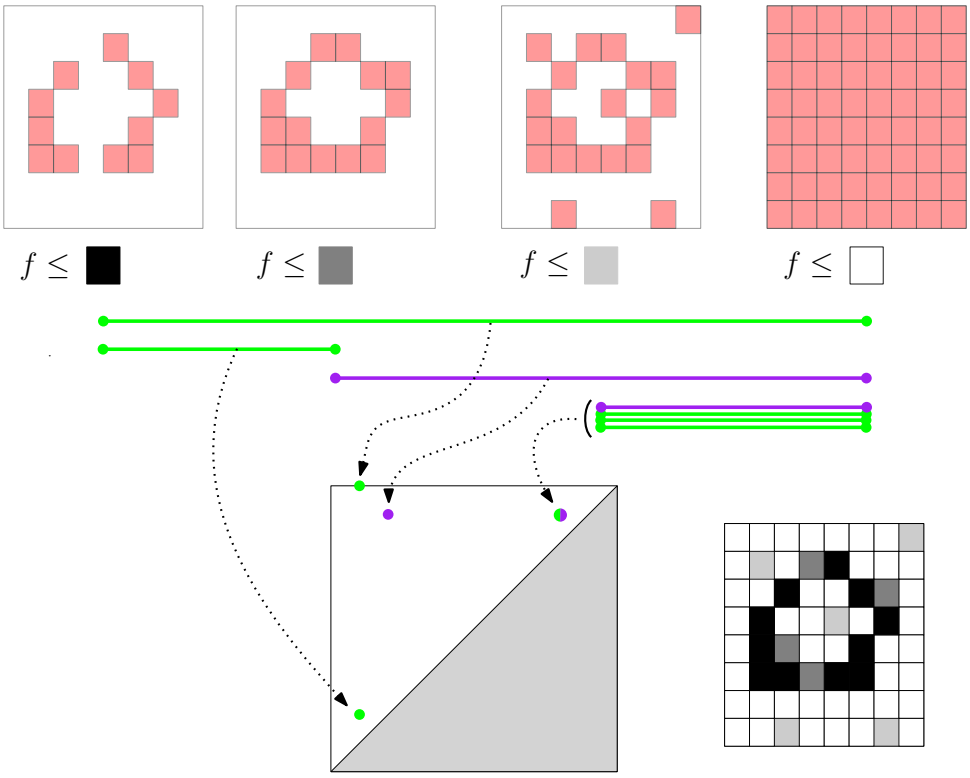

Figure 1.15: Another example of a persistence diagram construction with the sublevel sets of the grey value function defined on a blurry image of a zero.

One of the reasons why persistence diagrams are useful descriptors is that, in addition to be invariant to continuous deformations (that do not involve tearing or gluing), they are *stable* [42, 54]. Indeed, if persistence diagrams are computed with sublevel sets of similar functions, then the distance between them is upper bounded by the difference between the functions in the sup norm:

$$d_{\mathrm{b}}(\mathrm{Dg}(f), \mathrm{Dg}(g)) \leq \|f - g\|_\infty,$$

where d_{b} stands for the bottleneck distance between persistence diagrams, which is the cost of the best partial matching that one can find between the points of the persistence diagrams. This means that, for instance, if the positions of the images in Figure 1.14 are slightly perturbed, or if the blurry image of a zero in Figure 1.15 is slightly changed, then the resulting persistence diagrams will end up very close to the original ones in the bottleneck distance.

Persistence diagrams have proven useful in many data analysis applications, ranging from 3D shape analysis [38, 43] to glass material transition [73, 84] to genomics [24, 39], to name a few.

Mapper. As explained above, persistence diagrams summarize the topological information in data. However, they lose a lot of geometric information in the process: it is easy to build different spaces with

the same persistence diagrams. The *Mapper*[6], which was introduced in [129], is a direct approximation of the underlying object. It encompasses not only the topological features, but also additional information, on how the features are positioned with respect to each other for instance. As with persistence diagrams, a continuous scalar-valued function, sometimes called *filter*, is needed, as well as a cover of its image with open overlapping intervals. The idea is to compute the preimages by f of all intervals in the cover, to apply clustering on these preimages in order to refine them into connected components, and finally to link the connected components if they contain data points in common.

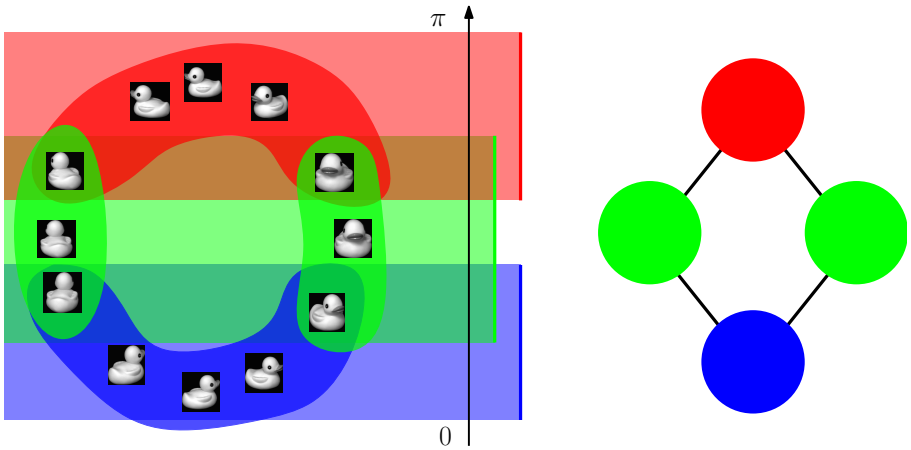

Figure 1.16: Example of Mapper computed on the point cloud of images with the angle function and a cover of three intervals.

We provide an example in Figure 1.16, where we consider again the point cloud of images. The continuous scalar-valued function is the absolute value of the angle at which the picture was taken, and its image $[0, \pi]$ is covered by three intervals (blue, green and red). In the preimage of the red and blue intervals there is just one connected component, whereas there are two in the preimage of the green interval. We obtain the Mapper by putting edges between the connected components according to whether they share data points or not; for instance, the green and blue or green and red connected components are linked whereas the red and blue are not. The Mapper has the homology of the circle, and is a direct approximation of the underlying support of the point cloud.

Note that the lengths of the intervals in the cover directly control the scale at which the data is observed: if the intervals are very small, the Mapper will have many disconnected nodes since the preimages of the intervals will contain at most one point. On the opposite, if the intervals have large lengths, the Mapper will have only few nodes since the preimages of the intervals are going to contain many points.

In practice, the Mapper has two major applications. The first one is data visualization and clustering. Indeed, when the cover $\mathcal{I}$ is minimal, the Mapper provides a visualization of the data in the form of a graph whose topology reflects that of the data. As such, it brings additional information to the usual clustering algorithms about the internal structure of the clusters, by identifying *flares* and *loops* that outline potentially remarkable topological information in the various clusters. See e.g. [138, 97, 125, 83] for examples of applications. The second application of Mapper deals with feature selection. Indeed, each feature of the data can be evaluated on its ability to discriminate the topological features mentioned above (flares, loops) from the rest of the data, using for instance Kolmogorov-Smirnov tests. See e.g. [97, 109, 122] for examples of applications.

1.2.3 Main bottlenecks

Even though Mapper and persistence diagrams enjoy many desireable properties, several limitations hinder their effective use in practice, in particular the *difficulty to set the parameters for Mapper* and the *non linearity* of the space of persistence diagrams.

[6]In this thesis, we call *Mapper* the mathematical object, not the algorithm used to build it.

Distances and stability for Mappers and Reeb graphs. One problem with the Mapper is that, contrary to the persistence diagrams, it has a parameter, which is the cover, and it is unclear how this cover should be tuned beforehand. Because of this, the Mapper seems to be a very *unstable* construction: it may happen that Mappers computed on nearby point clouds, as in Figure 1.17, or with similar covers, as in Figure 1.18, end up being very different.

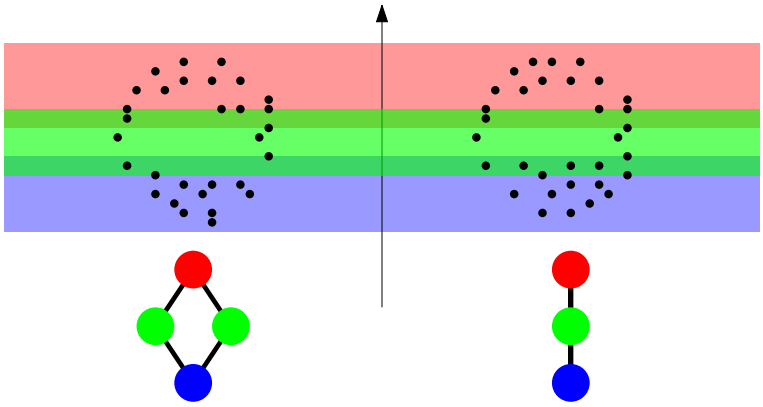

Figure 1.17: Mappers of two similar samplings of the circle, computed with the height function and a cover with three intervals.

This major drawback of Mapper is an important obstacle to its use in exploratory data analysis with non trivial datasets. The only answer proposed to this drawback in the literature consists in selecting parameters in a range of values for which the Mapper seems to be stable—see for instance [109].

Hence, deriving a stability theorem for Mappers would require to compare them with a metric that depends at least on the cover. Unfortunately, even though theoretical metrics can be defined, see e.g. [105], a computable and interpretable metric between Mappers is still lacking in the literature. To tackle this problem, one can take inspiration from another class of descriptors, which are very similar to Mappers: the *Reeb graphs*.

Reeb graphs. Note that the Mapper construction was originally defined for point clouds, but it can straightforwardly be extended to possibly non discrete topological spaces, for which clustering is not needed to compute the connected components of preimages. In that case, making the lengths of cover intervals go to zero leads to a limit object called the *Reeb graph*. Hence, Mappers (computed on non discrete topological spaces) are nothing but *approximations*, or *pixelized versions* of Reeb graphs, as illustrated in Figure 1.19.

This observation is important since several natural metrics enjoying stability properties already exist for Reeb graphs [7, 8, 61] and can be extended to Mappers. However, these metrics are not computable and thus cannot be used as is in practice [2]. Hence, there is an open question about how to define metrics for Mappers which would be both computable and stable.

Non linearity of the space of persistence diagrams. Even though persistence diagrams are stable descriptors, they cannot be plugged systematically into Machine Learning algorithms. Indeed, a large class of these algorithms require the data to lie either in Euclidean space (such as random forests), or at least in a Hilbert space (such as SVM). However, the space of persistence diagrams, equipped with the bottleneck distance, is neither Euclidean nor Hilbert. Even Fréchet means are not well-defined [136]. Fortunately, the *kernel trick* allows us to handle this kind of data. Assuming data points lie in some metric space (X, d_X), the kernel trick only requires a positive semi-definite function, called a *kernel*. This is a function $k : X \times X \to \mathbb{R}$ such that, for any $a_1, \cdots, a_n \in \mathbb{R}$ and $x_1, \cdots, x_n \in X$:

$$\sum_{i,j} a_i a_j k(x_i, x_j) \geq 0.$$

Due to Moore-Aronszajn's theorem [4], kernel values can be proven equal to the evaluation of the scalar product between embeddings of data into a specific Hilbert space that depends only on k and is generally

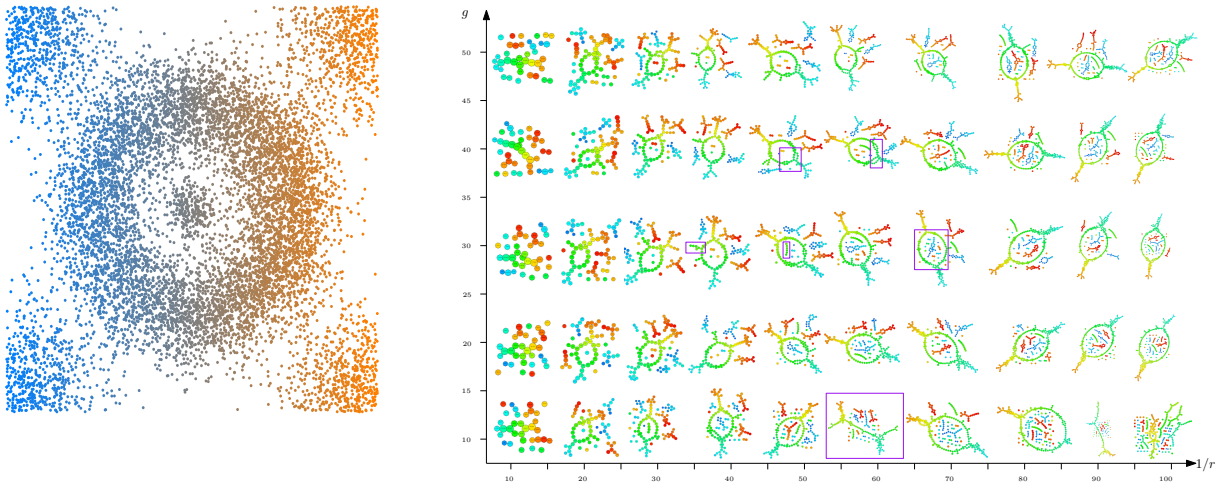

Figure 1.18: A collection of Mappers of the crater dataset computed with various covers (r is the length of the intervals and g is their overlap percentage) and the horizontal coordinate. Left: crater dataset colored with function values, from blue to orange. Right: Mappers computed with various parameters. The purple squares indicate topological features that suddenly appear and disappear in the Mappers.

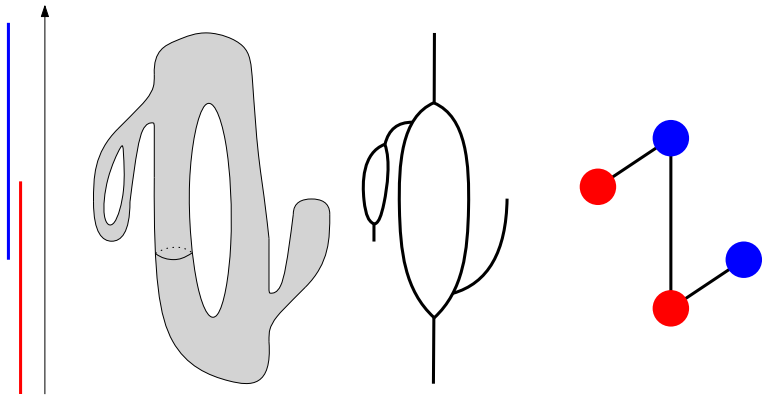

Figure 1.19: A surface embedded in $\mathbb{R}^3$ (left), its Reeb graph (middle) computed with the height function and its Mapper (right) computed with the height function and a cover with two intervals.

unknown. More formally, there exists a Hilbert space $\mathcal{H}_k$ such that, for any $x, y \in X$:

$$k(x, y) = \langle \Phi_k(x), \Phi_k(y) \rangle_{\mathcal{H}_k},$$

where the embedding Φ_k is called the *feature map* of k. Kernel values can thus be seen as generalized scalar products between data points, and can be directly plugged into Machine Learning algorithms. Hence, in our case, the question becomes that of finding kernels for persistence diagrams.

A common way to define kernels for points lying in a metric space (X, d_X) is to use *Gaussian* functions:

$$k_\sigma(x, y) = \exp\left(-\frac{d_X(x, y)}{2\sigma^2}\right),$$

where $\sigma > 0$ is a bandwidth parameter. A theorem of Berg et al. [11] shows that k_σ is a kernel, i.e. positive semi-definite, for all $\sigma > 0$ if and only if d_X is *negative semi-definite*, meaning that $\sum_{i,j} a_i a_j d_X(x_i, x_j) \leq 0$, for any $x_1, \cdots, x_n \in X$ and $a_1, \cdots, a_n \in \mathbb{R}$ such that $\sum_{i=1}^n a_i = 0$. Unfortunately, as shown by Reininghaus et al. [119], the bottleneck distance d_{b} for persistence diagrams is not negative semi-definite. Actually, one can build counter examples even for Wasserstein distances, which is another widely used class of distances. Hence, the use of Gaussian-type kernels for persistence diagrams is not possible with their canonical metrics.

Nevertheless, several kernels for persistence diagrams have been proposed in the last few years [1, 20, 90, 120], all of them enjoying stability properties upper bounding the distance between the embeddings of the persistence diagrams by the bottleneck or the Wasserstein distances between the diagram themselves. Hence, the metric distortion

$$\mathrm{dist}(\mathrm{Dg}, \mathrm{Dg}') = \frac{\|\Phi_k(\mathrm{Dg}) - \Phi_k(\mathrm{Dg}')\|_{\mathcal{H}_k}}{d_{\mathrm{b}}(\mathrm{Dg}, \mathrm{Dg}')}$$

is in general upper bounded. However, it is unclear whether it is also non-trivially lower bounded or not: it may happen that the embeddings of very different persistence diagrams actually lie very close to each other, which is not desirable for the discriminative power of the kernel. Think for instance of a constant kernel embedding: all persistence diagrams are mapped to the same element of a specific Hilbert space. This embedding is stable since the pairwise distances in the Hilbert space are all zero, but of course the kernel's results are very poor when plugged into Machine Learning algorithms. More generally, little is known concerning the behaviour and the properties of metrics of Hilbert spaces induced by kernels for persistence diagrams, and it remains an open question whether theoretical results on the discriminative power of kernels can be stated and proved or not.

1.2.4 Contributions

In this thesis, we investigate three problems: the interpretation of the topological features (i.e. with confidence regions) of the Mapper, the tuning of its parameters, and the global integration of topological descriptors into the framework of Machine Learning.

Distance between Reeb graphs. In Chapter 3, we define a computable pseudometric between Reeb graphs by comparing their persistence diagrams. Even though this distance is only a pseudometric, we are able to show that it is *locally equivalent* to other metrics. This local equivalence is then used to study the metric properties of the space of Reeb graphs when equipped with derived *intrinsic metrics*: we prove that all such intrinsic metrics are *strongly equivalent*, thus encompassing all approaches to compare Reeb graphs into a single framework. This work has been published in the proceedings of the Symposium on Computational Geometry 2017 [36].

Structure of the Mapper. In Chapter 4, we provide a link between the persistence diagrams of the Reeb graph and those of the Mapper (computed on the same topological space). Specifically, we show that the persistence diagram of the Mapper is obtained by removing specific points from the persistence diagram of the Reeb graph, namely those that lie in certain areas of the plane that only depend on the cover used to compute the Mapper. This explicit relation allows us to extend the computable pseudometric between Reeb graphs to Mappers. We then show that this distance *stabilizes* the Mapper, i.e. we provide a stability theorem for Mappers compared with this distance. This work has been published in the proceedings of the Symposium on Computational Geometry 2016 [35] and another version has been submitted to the Journal of Foundations of Computational Mathematics [34].

Discrete setting. In Chapter 5, we extend the previous theoretical results to the case where Mappers are computed on point clouds, and where connected components are computed with single-linkage clustering. Indeed, we provide sufficient conditions for which the Mapper computed on a point cloud coincides with the one computed on the (non discrete) support. Moreover, we show that the Mapper computed on the sampling of a topological space converges to the corresponding Reeb graph with an *optimal* rate of convergence, i.e. no other estimator of the Reeb graph can converge faster. Finding Mapper parameters for which the rate of convergence is optimal even allows us to provide *heuristics on the choice of these parameters*. These heuristics rely on bootstrap and only depend on the number of points in the sampling. We also provide a way to compute *confidence regions* for the various topological features of the Mapper. This work has been submitted to the Journal of Machine Learning Research [33].

Kernel methods. In Chapter 6, we apply Machine Learning to topological descriptors. Since Reeb graphs and Mappers are compared using their persistence diagrams, we focus on finding kernels for persistence diagrams.

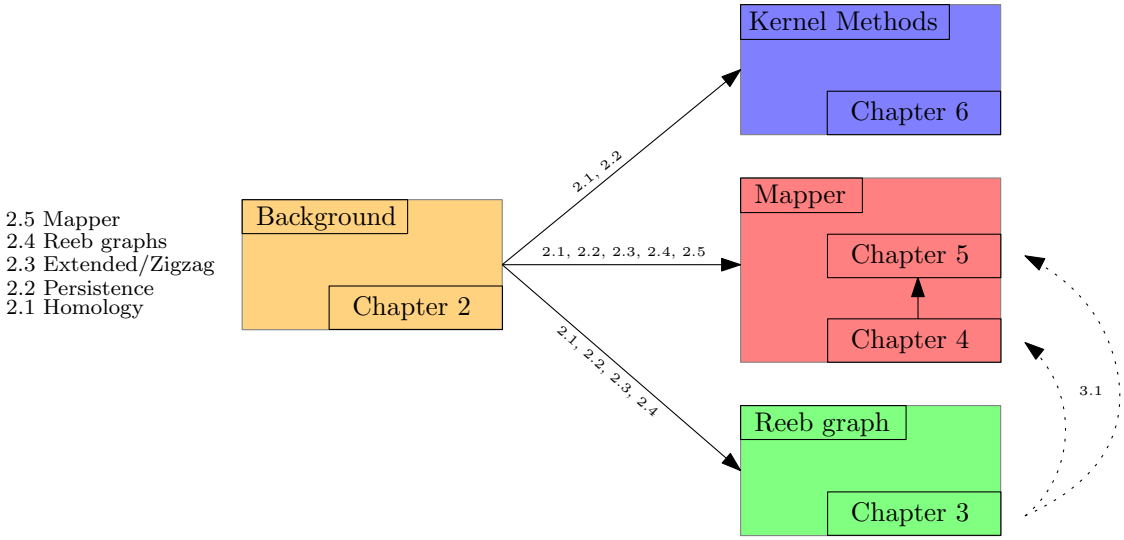

Figure 1.20: Plain arrows indicate dependence between chapters, and dotted arrows indicate partial dependence, meaning that only a small, skippable part of the chapter depends on the other.

We first define a *Gaussian-type kernel* by using a modification of the Wasserstein distance, called the *Sliced Wasserstein distance*. Indeed, we show that this distance, contrarily to the original Wasserstein distance, is actually negative semi-definite, and thus enables us to define a Gaussian kernel out of it. Morevover, we prove that the induced distance between persistence diagrams is *equivalent* to the original Wasserstein distance. Hence, this kernel, in addition to be stable and Gaussian, is also theoretically discriminative. We provide empirical evidence of this, showing significant improvements over the state-of-the-art kernels for persistence diagrams in a range of applications. This work has been published in the proceedings of the International Conference on Machine Learning 2017 [32].

Finally, we also provide a *vectorization method* to map persistence diagrams to $\mathbb{R}^D$, $D \in \mathbb{N}^*$. This provably stable mapping, even though not being injective, enables the use of persistence diagrams in algorithms and problems where Euclidean vectors are required. We detail an application example in which such structure is needed, namely 3D shape processing, for which we demonstrate that persistence diagrams are useful descriptors that provide additional information to the other usual descriptors. This work has been published in the proceedings of the Symposium on Geometry Processing 2015 [38].

How to read this thesis? This thesis is composed of four different parts:

- The first part is Chapter 2, in which we provide theoretical foundations for homology, persistence, Reeb graphs and Mapper. We also detail two extensions of persistence called *extended persistence* and *levelset zigzag persistence*.
- The second part is Chapter 3, which deals with Reeb graphs and their distances.
- The third part is composed of Chapters 4 and 5, which are about Mapper.
- The fourth part is Chapter 6. It is about defining kernels for persistence diagrams, in both finite and infinite dimensional Hilbert spaces.

See Figure 1.20. Chapter 2 contains the necessary background. The other chapters are contributions of this thesis. Chapters 3 and 4 have a strong topological flavor, while Chapter 5 has a statistical flavor and Chapter 6 is more oriented towards Machine Learning. These chapters are not independent, as illustrated in Figure 1.20, but the principal results and contributions are summarized at the beginning of each chapter. Hence, for each chapter, the reader can read either only the introduction, or the full chapter, depending on its personal background and interests.

CHAPTER 2

BACKGROUND ON TOPOLOGY

In this chapter, we review the basics of homology, persistent homology, Reeb graphs and Mappers. These descriptors are at the core of Topological Data Analysis, which heavily relies on their good properties, such as their stability with respect to perturbations of the data—see Theorems 2.2.15, 2.3.1 and 2.4.10.

Plan of the Chapter. We introduce homology in Section 2.1, and persistence theory in Section 2.2. We then present two extensions of persistence that we use in this thesis, namely *extended* and *levelset zigzag* persistence in Section 2.3. We finally provide background on Reeb graphs in Section 2.4 and Mappers in Section 2.5.

2.1 Homology Theory

Homology is the main building block of persistence theory. In this section, we first review simplicial homology, and then two extensions thereof, namely singular and relative homology. We refer the interested reader to [106] for more details.

2.1.1 Simplices and Simplicial Complexes

We start with the definition of *abstract simplices* and *abstract simplicial complexes.*

Definition 2.1.1. *Let E be a finite index set. An* abstract simplex *σ is an element of $\mathcal{P}(E)$. Its elements are called its* vertices*. When σ has a finite number of vertices, we write it: $\sigma = \{v_0, v_1, \cdots, v_p\}$, $p \in \mathbb{N}$. An* abstract simplicial complex *K is a non-empty subset of $\mathcal{P}(E)$ such that $\forall \sigma \in K$, $\tau \subseteq \sigma \Rightarrow \tau \in K$. In particular, $\emptyset \in K$.*

Dimension, faces and skeletons. The different sets in K are the *simplices* of K. The *dimension* of a simplex σ is $\dim(\sigma) = \mathrm{card}(\sigma) - 1$, and the dimension of a simplicial complex K is $\dim(K) = \max_{\sigma \in K} \dim(\sigma)$. For a given simplex σ, a *p-face* of σ is a subset τ of σ of dimension p. Thus, according to Definition 2.1.1, all the faces of any simplex in the complex must also be in the complex. The union of the p-dimensional simplices of every simplex in K gives the *p-skeleton* of K. The 0-skeleton of K, i.e. its set of vertices, is denoted by $V(K)$.

Orientations. We define equivalence classes for the orderings of the vertices of a simplex σ in the following way: two orderings of its vertices are equivalent if and only if they differ from one another by an even permutation. This leads to two equivalence classes for the orderings of the vertices of σ, also called two *orientations* of σ. When the simplex is oriented, we write: $\sigma = [v_0, v_1, \cdots, v_p]$ to specify the equivalence class of the particular ordering $v_0, v_1, \cdots, v_p$.

Definition 2.1.2. *A* geometric realization ψ *in* $\mathbb{R}^D$*,* $D \in \mathbb{N}^*$*, of an abstract simplex* σ *of dimension* p*, is the convex hull in* $\mathbb{R}^D$ *of the point set* $\{\psi(v_0), \cdots, \psi(v_p)\}$*:*

$$\psi(\sigma) = \left\{ \sum_{i=0}^{p} \lambda_i \psi(v_i) \ : \ \sum_{i=0}^{p} \lambda_i = 1, \ \lambda_i \geq 0 \right\},$$

where the points $\psi(v_0), \cdots, \psi(v_p)$ *are affinely independent.*

Obviously, a geometric realization is not unique and is not possible for every dimension D. In particular, we must have $p \leq D$. The geometric realization ψ of an abstract simplex σ of dimension p is a *geometric simplex of dimension* p.

Definition 2.1.3. *A* geometric realization ψ *in* $\mathbb{R}^D$*,* $D \in \mathbb{N}$*, of an abstract simplicial complex* K *maps every vertex of* $V(K)$ *to a point in* $\mathbb{R}^D$*, so that the two following properties are satisfied:*

- *for every abstract simplex* σ *in* K*,* $\psi(\sigma)$ *is a geometric simplex,*
- *for any two distinct simplices* σ_1 *and* σ_2 *in* K*,* $\psi(\sigma_1) \cap \psi(\sigma_2) = \psi(\sigma_1 \cap \sigma_2)$*, with the convention that* $\psi(\emptyset) = \emptyset$*.*

The geometric realization ψ of an abstract simplicial complex K of dimension p is a *geometric simplicial complex of dimension* p. In general, we write $|K|$ to denote a geometric realization of K.

Examples. In $\mathbb{R}^3$, we can have 0, 1, 2 and 3-dimensional geometric simplices, respectively points, segments, triangles and tetrahedra—see Figure 2.1. See also Figure 2.2 for an example of geometric simplicial complex.

Minimal dimension. The following theorem, whose proof relies on simple codimension considerations, states what is the minimum value for D so that a geometric realization of K is always possible:

Theorem 2.1.4 (Whitney's Embedding Theorem)**.** *Any abstract simplicial complex* K *of dimension* $n \in \mathbb{N}$ *has a generic geometric realization in* $\mathbb{R}^{2n+1}$*.*

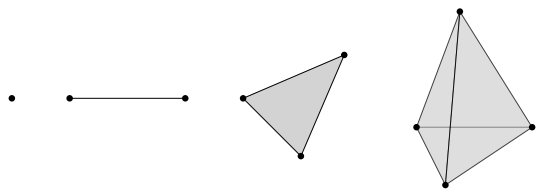

Figure 2.1: Example of geometric simplices in dimension 0, 1, 2 and 3.

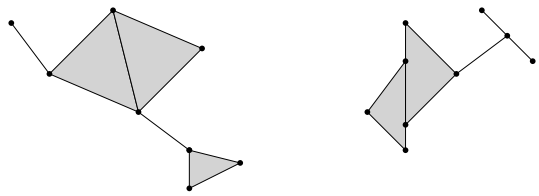

Figure 2.2: The complex on the right hand side is not a geometric simplicial complex, because the intersection of the two triangles should be empty, as the triangles do not share any vertex, whereas it is not. The complex on the left hand side is a simplicial complex.

2.1.2 Simplicial Homology

The definition of homology is based on *p-chains*, i.e. formal sums of simplices.

Definition 2.1.5. *The set of p-chains of K, denoted by $C_p(K;\mathbb{Z})$, is the free abelian group generated by the oriented p-simplices of K.*

In practice, we often work with coefficients in a field, like $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z}$ (if q is a prime integer).

Definition 2.1.6. *Let σ be an oriented simplex of dimension p. The* boundary *of σ is the $(p-1)$-chain given by the alternate sum of all of the oriented $(p-1)$-faces of σ. Formally, if $\sigma = [v_0, \cdots, v_p]$, the boundary of σ is:*

$$\sum_{i=0}^{p} (-1)^i [v_0, \cdots, \hat{v}_i, \cdots, v_p],$$

where $[v_0, \cdots, \hat{v}_i, \cdots, v_p]$ is the oriented $(p-1)$-face of σ, with missing v_i.

By linearity, we extend the definition of the boundary to a p-chain of K. By convention, the boundary of a 0-chain is 0. The resulting *boundary operator* $\partial_p : C_p(K;\mathbb{Z}) \to C_{p-1}(K;\mathbb{Z})$ sends a p-chain $c = \sum_i n_i \sigma_i$ to its boundary—see Figure 2.3:

$$\partial_p(c) = \sum_i n_i \partial_p(\sigma_i).$$

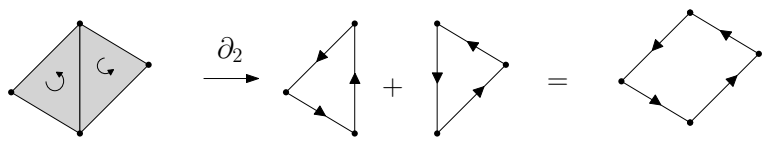

Figure 2.3: Action of the boundary operator on the sum of two oriented simplices. The middle edge cancels as it is counted twice with opposite orientations.

Definition 2.1.7. *A* p-cycle *is a p-chain whose boundary is 0. The subgroup of all p-cycles is* $\ker(\partial_p)$.

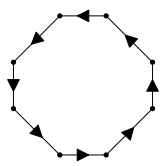

Figure 2.4: Example of an oriented 1-cycle.

Let us state the main property of the boundary operator:

Proposition 2.1.8. $\forall p \in \mathbb{N}^*,\ \partial_{p-1} \circ \partial_p = 0$.

Hence, we can extend this result to p-chains by linearity: if $c = \sum_i n_i \sigma_i$ then $\partial_{p-1} \circ \partial_p(c) = 0$. This property allows to define a *chain complex*.

Definition 2.1.9. *A* chain complex *$\mathcal{C}$ is a family of abelian groups C_p, $p \in \mathbb{N}^*$, together with homomorphisms $\phi_p : C_p \to C_{p-1}$ such that $\phi_{p-1} \circ \phi_p = 0$.*

In particular, the family of chain groups of a simplicial complex K together with the boundary operators is a chain complex. In Sections 2.1.3 and 2.1.4, we build other examples of chain complexes. Since $\mathrm{im}(\phi_p) \subseteq \ker(\phi_{p-1})$, we can define the *pth-homology group* of a chain complex as the quotient of those spaces.

Definition 2.1.10. *The pth-homology group of a chain complex $\mathcal{C}$ is:*

$$H_p(\mathcal{C}) = \ker(\phi_p)/\mathrm{im}(\phi_{p+1}).$$

In particular, the *simplicial pth-homology group* of a simplicial complex K is:

$$H_p(K;\mathbb{Z}) = \ker(\partial_p)/\mathrm{im}(\partial_{p+1}).$$

A p-cycle in $\mathrm{im}(\partial_{p+1})$ is said to be *trivial* and two equivalent p-cycles modulo $\mathrm{im}(\partial_{p+1})$ are said to be *homologous*. If K is a finite simplicial complex, then $C_p(K;\mathbb{Z})$ is finitely generated (by the p-simplices of K), and so is $H_p(K;\mathbb{Z})$.

Theorem 2.1.11 (Decomposition of finitely generated abelian groups.)**.** *Every finitely generated abelian group G is isomorphic to a direct sum of the form: $\mathbb{Z}^n \oplus \mathbb{Z}_{q_1} \oplus \cdots \oplus \mathbb{Z}_{q_m}$, where $q_1, \cdots, q_m$ are powers of prime numbers. The integer n is called the rank of G, and $\oplus_{i=1}^m \mathbb{Z}_{q_i}$ is called the torsion subgroup of G.*

Definition 2.1.12. *Let K be a finite simplicial complex. The* pth-Betti number $\beta_p(K;\mathbb{Z})$ *of K, $p \in \mathbb{N}$, is the rank of $H_p(K;\mathbb{Z})$.*

Interpretation. If we work with coefficients in $\mathbb{Z}_2$, then the Betti numbers $\beta_0(K,\mathbb{Z}_2)$, $\beta_1(K,\mathbb{Z}_2)$ and $\beta_2(K,\mathbb{Z}_2)$ can be interpreted as respectively the number of connected components of K, the number of holes in K and the number of cavities, or voids, in K. To convince oneself, let us look at the 0-dimensional case. A 0-chain is a set of vertices of a simplicial complex. We claim that each of these vertices corresponds to a specific connected component. Indeed, if we select an arbitrary vertex in every connected component then every vertex of a given connected component is homologous to the corresponding arbitrary vertex. The proof is immediate: if two vertices v_1 and v_2 are in the same connected component, there exists a path of edges, or a 1-chain, between them. If we compute the boundary of this path, we get the boundary of every edge in the path, which consists of two vertices. As the edges in the path are linked, all the vertices will be counted twice and thus will disappear (since the field of coefficients is $\mathbb{Z}_2$), except for the vertices at the beginning and the end of the path, in other words v_1 and v_2, that are thus homologous. On the contrary, no such path exists if the vertices are not in the same connected component.

Example on the annulus. To make these notions clearer, let us look at a specific example, the annulus of Figure 2.5. As we said, β_0 is fairly easy to compute, it is equal to 1 in the example. Let us now look at β_1. We recall that in $\mathbb{Z}_2$, we do not consider orientations or alternate sums. Clearly, $\{a_0, a_1\} + \{a_1, a_2\} + \{a_2, a_0\}$ is a 1-cycle because:

$$\begin{aligned}
&\partial_1(\{a_0, a_1\} + \{a_1, a_2\} + \{a_2, a_0\}) \\
&\quad = \partial_1(\{a_0, a_1\}) + \partial_1(\{a_1, a_2\}) + \partial_1(\{a_2, a_0\}) \\
&\quad = \{a_0\} + \{a_1\} + \{a_1\} + \{a_2\} + \{a_2\} + \{a_0\} = 0
\end{aligned}$$

This cycle is also non trivial. Every other 1-cycle is homologous, for instance let us consider $\{a_0, a_1\} + \{a_1, a_3\} + \{a_3, a_2\} + \{a_2, a_0\}$, which is also non trivial. Then their sum is $\{a_1, a_3\} + \{a_3, a_2\} + \{a_2, a_1\}$, which is trivial (it is the boundary of $\{a_1, a_2, a_3\}$). Thus $\beta_1 = 1$.

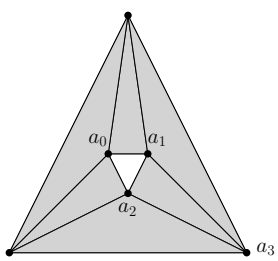

Figure 2.5: Example of simplicial complex representing an annulus. The Betti numbers are $\beta_0 = \beta_1 = 1$.

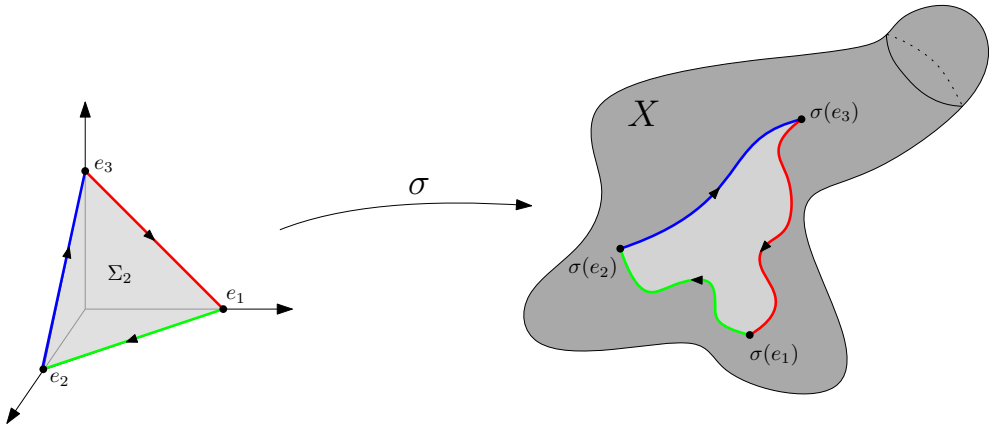

Figure 2.6: Oriented singular 2-simplex σ on a 3D surface X.

Morphisms between homology groups. Given several chain complexes, morphisms between their corresponding homology groups can be built from *chain maps*.

Definition 2.1.13. *Let* $\mathcal{C} = \cdots \xrightarrow{\phi_{p+1}} C_p \xrightarrow{\phi_p} C_{p-1} \xrightarrow{\phi_{p-1}} \cdots$ *and* $\mathcal{C}' = \cdots \xrightarrow{\phi'_{p+1}} C'_p \xrightarrow{\phi'_p} C'_{p-1} \xrightarrow{\phi'_{p-1}} \cdots$ *be chain complexes. A family of homomorphisms* $f = \{f_p : C_p \to C'_p\}_{p\in\mathbb{N}}$ *is a* chain map *if*

$$\phi'_p \circ f_p = f_{p-1} \circ \phi_p,$$

for any $p \in \mathbb{N}^*$.

Proposition 2.1.14. *A chain map* $f : \mathcal{C} \to \mathcal{C}'$ *induces a homomorphism* $f_* : H_*(\mathcal{C}) \to H_*(\mathcal{C}')$. *Moreover,* (i) *the identity map* id *of* $\mathcal{C}$ *is a chain map, and* id_* *is the identity map of* $H_*(\mathcal{C})$, *and* (ii) *if* $f : \mathcal{C} \to \mathcal{C}'$ *and* $g : \mathcal{C}' \to \mathcal{C}''$ *are chain maps, then* $g \circ f : \mathcal{C} \to \mathcal{C}''$ *is a chain map and* $(g \circ f)_* = g_* \circ f_*$.

Morphisms between simplicial homology groups. Chain maps between simplicial homology groups arise from *simplicial maps* between simplicial complexes.

Definition 2.1.15. *Let* K, L *be two abstract simplicial complexes. A map* $f : V(K) \to V(L)$ *is a* simplicial map *if* $\{v_0, \cdots, v_p\} \in K \Rightarrow \{f(v_0), \cdots, f(v_p)\} \in L$.

Proposition 2.1.16. *Let* K, L *be two abstract simplicial complexes, and* $f : V(K) \to V(L)$ *be a simplicial map. Then,* f *induces a chain map between the chain complexes* $\{C_p(K;\mathbb{Z}), \partial_p\}_{p\in\mathbb{N}}$ *and* $\{C_p(L;\mathbb{Z}), \partial_p\}_{p\in\mathbb{N}}$.

2.1.3 Singular Homology

Other chain complexes can be defined if the space under consideration is not a simplicial complex. This can be done using the so-called *singular homology*. Intuitively, singular simplices are images of usual simplices under continuous functions. We refer the interested reader to [106] for further details.

Definition 2.1.17. *Let* $\Sigma_p = \{v_1, \cdots, v_{p+1}\}$ *be the* oriented standard p-simplex, *i.e. the geometric simplex in* $\mathbb{R}^\infty$ *whose vertices are defined by* $v_i = e_i$, *where* e_i *is the* i*th element of the standard basis of* $\mathbb{R}^\infty$, *together with the orientation induced by the basis ordering. Let* X *be a topological space. An* oriented singular p-simplex *of* X *is the image of* Σ_p *under a continuous mapping* $\sigma : \Sigma_p \to X$, *together with an orientation induced by the one of* Σ_p. *We write* $\sigma([v_1, \cdots, v_{p+1}])$ *to denote such a simplex together with its orientation.*

Note that σ need not be injective. It may be the constant map for instance. We give an example of an oriented singular 2-simplex in Figure 2.6.

Singular homology groups. All the definitions in the previous section extend almost directly. The group of *singular p-chains* is defined as the free abelian group generated by the oriented singular p-simplices of X. Note that this group may contain uncountably many generators. The *(singular) boundary operator* ∂_p^{sing} is defined on a singular simplex as:

$$\partial_p^{\text{sing}}(\sigma([v_1, \cdots, v_{p+1}])) = \sum_{i=1}^{p+1} (-1)^i \sigma_i([v_1, \cdots, \hat{v}_i, \cdots, v_{p+1}]),$$

where σ_i is the restriction of σ on the $(p-1)$-face of Σ_p induced by the removal of v_i. It is then extended by linearity to singular chains, and we have again $\partial_p^{\text{sing}} \circ \partial_{p+1}^{\text{sing}} = 0$, so we can define a chain complex with these boundary operators. Hence, we define the *singular pth-homology group* of X as $H_p(X;\mathbb{Z}) = \ker(\partial_p^{\text{sing}})/\text{im}(\partial_{p+1}^{\text{sing}})$, i.e. the group of those singular p-chains with null boundary (also called *singular p-cycles*) that are not images by ∂_{p+1} of singular $(p+1)$-chains.

Singular and simplicial homologies. Given an abstract simplicial complex K and a geometric realization $|K|$ thereof, one may ask for the relationships between the simplicial homology groups of K and the singular homology groups of $|K|$. It turns out that they are essentially the same:

Proposition 2.1.18 (§34 in Chapter 4 in [106])**.** *Let K be an abstract simplicial complex and let $|K|$ be a geometric realization of K. Then, the singular homology groups of $|K|$ and the simplicial homology groups of K are isomorphic.*

Morphisms between singular homology groups. There is an easy way to build chain maps between the chain complexes induced by the singular chain groups of two spaces X and Y. Indeed, given a function $f : X \to Y$, continuity is a sufficient requirement to build such a chain map.

Proposition 2.1.19 (Theorems 29.1 and 29.2 in [106])**.** *Let X, Y be topological spaces and $f : X \to Y$ be a continuous function. Then, f induces a chain map between the chain complexes $\{C_p(X;\mathbb{Z}), \partial_p\}_{p\in\mathbb{N}}$ and $\{C_p(Y;\mathbb{Z}), \partial_p\}_{p\in\mathbb{N}}$.*

Invariance to homotopy equivalence. One of the key properties of homology groups is their invariance to continuous deformations of spaces. To formalize this, we use the notion of *homotopy equivalence* between topological spaces.

Definition 2.1.20. *Let X, Y be topological spaces, and let $f, f' : X \to Y$ be continuous functions. The functions f and f' are said to be* homotopic *if there exists a continuous function $F : X \times [0,1] \to Y$ such that $F(\cdot, 0) = f$ and $F(\cdot, 1) = f'$. The spaces X and Y are said to have* the same homotopy type*, or to be* homotopy equivalent*, if there exist $f : X \to Y$ and $g : Y \to X$ such that $f \circ g$ is homotopic to id_Y and $g \circ f$ is homotopic to id_X.*

Note that if X and Y are homeomorphic, i.e. there exists a continuous bijection $f : X \to Y$ such that the inverse f^{-1} is also continuous, then they have the same homotopy type (it suffices to take $g = f^{-1}$ in Definition 2.1.20). Proposition 2.1.19 allows to show the following proposition, which states that homology is invariant under homotopy equivalences—and thus also under homeomorphisms.

Proposition 2.1.21. *Let X, Y be homotopy equivalent topological spaces. Then $H_*(X;\mathbb{Z}) \simeq H_*(Y;\mathbb{Z})$.*

In particular, if a topological space X is triangulable, then there is an abstract simplicial complex K such that X and $|K|$ are homeomorphic. Thus, the singular homology groups of X are isomorphic to the ones of $|K|$, which in turn are isomorphic to the simplicial homology groups of K.

2.1.4 Relative Homology

Relative homology groups are computed with pairs of topological spaces. As for singular homology, it is a simple extension of simplicial homology.

Definition 2.1.22. *Let $Y \subseteq X$ be topological spaces. Let $C_p(X;\mathbb{Z})$ and $C_p(Y;\mathbb{Z})$ be the groups of p-chains of X and Y respectively. Then, the group of* relative p-chains *is the quotient group:*

$$C_p(X, Y;\mathbb{Z}) = C_p(X;\mathbb{Z})/C_p(Y;\mathbb{Z}).$$

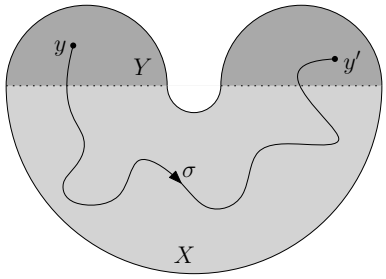

Figure 2.7: In this example, we study the pair of spaces $Y \subseteq X$. The oriented singular 1-simplex $\sigma = [y, y']$ has boundary $[y'] - [y] \in Y$. Hence, $\partial_1(\sigma) \neq 0$ while $\partial_1^{\text{rel}}(\sigma) = 0$: σ is a cycle in relative homology but not in usual homology.

Relative homology groups. The usual boundary operator commutes with the inclusion: letting $\iota_p : C_p(Y;\mathbb{Z}) \hookrightarrow C_p(X;\mathbb{Z})$ denote the canonical inclusion, we have $\partial_p \circ \iota_p = \iota_{p-1} \circ \partial_p$. Hence, ∂_p induces the *(relative) boundary operator* $\partial_p^{\text{rel}} : C_p(X,Y;\mathbb{Z}) \to C_{p-1}(X,Y;\mathbb{Z})$, which satisfies $\partial_p^{\text{rel}} \circ \partial_{p+1}^{\text{rel}} = 0$. Once again, this allows to define a chain complex, which in turn induces the so-called *relative pth-homology group* with $H_p(X,Y;\mathbb{Z}) = \ker(\partial_p^{\text{rel}})/\text{im}(\partial_{p+1}^{\text{rel}})$. Note that these definitions hold also for abstract simplicial complexes.

Example. It is easy to build examples where homology and relative homology differ. For instance, any p-chain included in Y is trivial in $C_p(X,Y;\mathbb{Z})$. It may also happen that p-chains of nonzero boundary with the usual boundary operator become p-cycles in relative homology. See Figure 2.7 for instance.

2.2 Persistence Theory

We now describe persistent homology for topological spaces. However, we recall from Proposition 2.1.18 that all definitions hold for simplicial complexes as well.

2.2.1 Filtrations

Intuitively, the aim of persistence is to study the evolution of the homology groups through a *filtration*.

Definition 2.2.1. *A* filtration *is an $\mathbb{R}$-indexed family of topological spaces $\{X_\alpha\}_{\alpha\in\mathbb{R}}$ that are nested with respect to inclusion, that is $s \leq t \Rightarrow X_s \subseteq X_t$.*

Note that, when the X_i are simplicial complexes, Definition 2.2.1 means that a simplex $\sigma_i \in X_i$ cannot appear in the filtration before its faces.

Definition 2.2.2. *Let $f : X \to \mathbb{R}$ be a continuous function defined on a topological space X. The filtration $\{F_\alpha\}_{\alpha\in\mathbb{R}}$* induced by *$f$ is the filtration composed of the sublevel sets of f: $F_\alpha = f^{-1}((-\infty, \alpha])$.*

One cannot choose just any function to build a filtration. For instance, when the spaces are simplicial complexes, the value of f on a simplex σ must be superior to its values on all the faces of σ, so that the faces of σ are included in the filtration before σ itself. We must have:

$$\forall \sigma \in X_i,\ \tau \text{ is a face of } \sigma \ \Rightarrow\ f(\tau) \leq f(\sigma).$$

A classical way to accomplish this is to define the values of f on $V(K)$ and to define f on simplices of dimension $p > 0$ in the following way: $f(\{v_0, \cdots, v_p\}) = \max\{f(v_0), \cdots, f(v_p)\}$. This is also known as the *lower-star filtration of f*. See Figure 2.8, where a function is defined on the 8 vertices of a simplicial complex K. The order of appearance of the simplices depends on the values of f at these vertices.

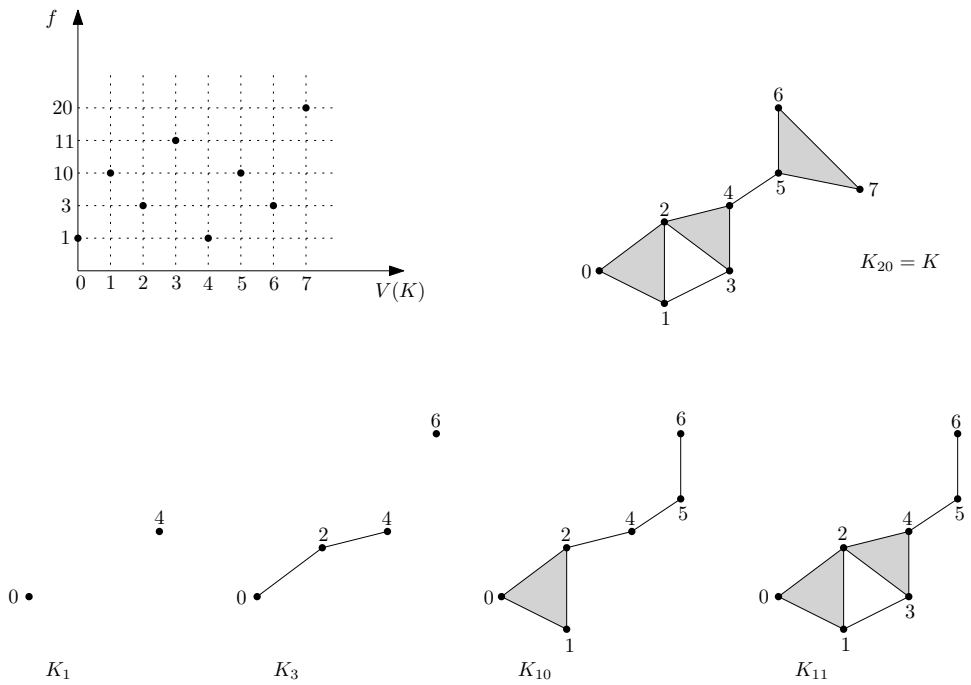

Figure 2.8: Example of lower star filtration of a function f

2.2.2 Persistence Modules

The main object of persistence theory is the so-called *persistence module.*

Definition 2.2.3. *Let $\mathbb{K}$ be a field. A* persistence module $\mathbb{U}$ *is a set of $\mathbb{K}$-vector spaces indexed by $\mathbb{R}$, denoted by $\{U_\alpha\}_{\alpha\in\mathbb{R}}$, together with a family of linear maps $\{u_\alpha^\beta : U_\alpha \to U_\beta\}_{\alpha,\beta\in\mathbb{R},\alpha\leq\beta}$ such that:*

- $\forall\alpha \in \mathbb{R}$, $u_\alpha^\alpha = \mathrm{id}_{U_\alpha}$ *and*
- $\forall\alpha \leq \beta \leq \gamma$, $u_\beta^\gamma \circ u_\alpha^\beta = u_\alpha^\gamma$.

Interval module. A notable example of persistence module is the *interval module* $\mathbb{I}_I$ on an interval $I \subseteq \mathbb{R}$, defined by: $(\mathbb{I}_I)_\alpha = \mathbb{K}$ if $\alpha \in I$ and 0 otherwise; $(i_I)_\alpha^\beta = \mathrm{id}_\mathbb{K}$ if $[\alpha, \beta] \subseteq I$ and 0 otherwise.

Persistence module of a filtration. Let $\mathbb{K}$ be a field, X a topological space, $\{X_s\}_{s\in\mathbb{R}}$ a filtration of X, $H_p(X_s;\mathbb{K})$ and $H_p(X_t;\mathbb{K})$ the pth-homology groups of X_s and X_t (with $s \leq t$ thus $X_s \subseteq X_t$). We can consider the inclusion maps $\iota_s^t = H_p(X_s;\mathbb{K}) \to H_p(X_t;\mathbb{K})$ induced by the canonical inclusion $X_s \hookrightarrow X_t$. Note that these maps depend on the homological dimension p and may not be injective. The *pth-persistence module of X associated to the filtration* $\{X_s\}_{s\in\mathbb{R}}$ is the persistence module $\{H_p(X_s;\mathbb{K}), \{\iota_s^t\}_{t\geq s}\}_{s\in\mathbb{R}}$.

Morphisms. The persistence modules can actually be seen as the objects of an abelian category, whose morphisms we now define.

Definition 2.2.4. *Let $\mathbb{U} = \{U_\alpha, u_\alpha^\beta\}$ and $\mathbb{V} = \{V_\alpha, v_\alpha^\beta\}$ be two persistence modules. A morphism Ψ between $\mathbb{U}$ and $\mathbb{V}$ is a family of morphisms $\Psi = \{\psi_\alpha : U_\alpha \to V_\alpha \,:\, \alpha \in \mathbb{R}\}$ such that for all $\alpha, \beta \in \mathbb{R}$ such that $\alpha \leq \beta$, we have $\psi_\beta \circ u_\alpha^\beta = v_\alpha^\beta \circ \psi_\alpha$. If every ψ_α is an isomorphism, then Ψ is called an* isomorphism, *and $\mathbb{U}$ and $\mathbb{V}$ are* isomorphic, *which we write: $\mathbb{U} \simeq \mathbb{V}$.*

Direct sum. The direct sum of two modules $\mathbb{W} = \mathbb{U} \oplus \mathbb{V}$ is the module defined by $W_\alpha = U_\alpha \oplus V_\alpha$ for all $\alpha \in \mathbb{R}$, and $w_\alpha^\beta = u_\alpha^\beta \oplus v_\alpha^\beta$ for all $\alpha, \beta \in \mathbb{R}$, such that $\alpha \leq \beta$. We say that a persistence module $\mathbb{U}$ is *decomposable* if it is isomorphic to the direct sum of two non zero modules.

Proposition 2.2.5 (Proposition 2.6 in [45]). *An interval module admits no other decomposition than the trivial one:* $\mathbb{I}_I = 0 \oplus \mathbb{I}_I = \mathbb{I}_I \oplus 0$.

A module $\mathbb{U}$ is said to be *decomposable into intervals* if it admits a decomposition composed of interval modules $\mathbb{U} \simeq \oplus_{I \in \mathcal{I}} \mathbb{I}_I$. This decomposition is unique up to isomorphism and reordering of the terms, as stated in the following proposition.

Proposition 2.2.6 (Theorem 2.7 in [45]). *Let* $\mathbb{U}$ *be a decomposable persistence module. Assume that there exists two different decompositions:* $\mathbb{U} \simeq \oplus_{I \in \mathcal{I}} \mathbb{I}_I \simeq \oplus_{J \in \mathcal{J}} \mathbb{I}_J$. *Then there exists a bijection* $b : \mathcal{I} \to \mathcal{J}$ *such that* $\mathbb{I}_I \simeq \mathbb{I}_{b(I)}$ *for all* $I \in \mathcal{I}$.

Theorem 2.2.7 (Theorem 1.13 in [113], Theorem 2.8 in [45]). *Let* $\mathbb{U} = \{U_\alpha, u_\alpha^\beta\}$ *be a persistence module. If* $\mathbb{U}$ *is* tame, *i.e.* u_α^β *has finite rank for any* $\alpha \leq \beta$, *then* $\mathbb{U}$ *is decomposable into intervals.*

Special case. In the case of simplicial complexes, Theorem 2.2.7 applies to the pth-persistence module of any filtration of any simplicial complex K, provided that K has a finite number of p-simplices.

Definition 2.2.8. *Let* X *be a topological space and* $f : X \to \mathbb{R}$ *be a continuous function. We say that* f *is* tame *if the persistence module induced by its sublevel sets is tame.*

It follows that the persistence module of any tame function is decomposable into intervals.

2.2.3 Persistence Diagram

When a persistence module is decomposable into intervals, for instance when it comes from the filtration induced by a tame function, it is convenient to plot each interval in the extended plane $\bar{\mathbb{R}}^2$, where $\bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$, using the endpoints of the intervals as coordinates. This way of encoding a persistence module is called a *persistence diagram.*

Definition 2.2.9. *Let* $\mathbb{U}$ *be a decomposable persistence module of the form*

$$\mathbb{U} \simeq \bigoplus_{\alpha \in A} \mathbb{I}_{(b_\alpha, d_\alpha)},$$

where A *is an index set,* b_α *and* $d_\alpha \in \bar{\mathbb{R}}$, *and* (b_α, d_α) *can be the open interval* (b_α, d_α), *the closed one, or one of the two half-closed ones. The* persistence diagram *of* $\mathbb{U}$ *is the multiset:* $\mathrm{Dg}(\mathbb{U}) = \sqcup_{\alpha \in A} \{(b_\alpha, d_\alpha)\}$.

Birth and death. b_α is called the birth time of homological feature α, and d_α is called its death time. Note that a point in a persistence diagram can encode several different homology generators. The number of generators this point represents is called the *multiplicity* of the point.

Example on a simplicial complex. We compute homology with coefficients in $\mathbb{Z}_2$. Let us consider the example of Figure 2.9. The final simplicial complex K is shown at the end of the discrete filtration, it has dimension 2 with $\beta_0 = 1$, $\beta_1 = 0$ and $\beta_2 = 0$. At time 0, there is nothing ($\beta_0 = \beta_1 = 0$). At time 1, a new connected component is born, thus an interval with birth time equal to 1 is created in H_0 (β_0 becomes 1). At time 2, three other connected components (two isolated vertices and one triangle) are born, three other intervals with birth 2 are created in H_0 (and $\beta_0 = 4$), but there is still no 1-cycle. At time 3, one of the connected components is merged with the first connected component, thus one of the previous interval with birth time 2 in H_0 has a death time set to 3. A 1-cycle appears too, an interval with birth 3 for H_1 is created (and $\beta_1 = 1$). At time 4, the 1-cycle is killed, the corresponding interval has a death time set to 4, as well as one of the two connected components added at time 2. Finally at time 5, the two remaining connected components are merged together: one of the still non closed intervals in H_0 has a death time set to 5 (the most recent one, i.e. the one with birth time 2) and the other to $+\infty$ (and $\beta_0 = 1, \beta_1 = 0$).

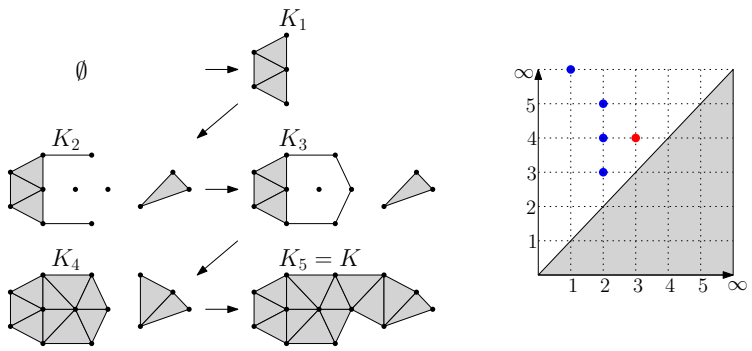

Figure 2.9: Example of a filtration and its corresponding persistence diagram (0-dimensional points are marked in blue and 1-dimensional points in red).

Algorithm 1: Computation of the persistence diagram

Input: $\{K_i\}_{1 \leq i \leq n}$.
for $i = 0, \cdots, n$ **do**
  while $\exists j < i$ such that $l_j == l_i \neq -1$ **do**
    $M_i = M_i + M_j$ in $\mathbb{Z}_2$
  end while
end for
Output: $\mathrm{Dg} = \sqcup_{i=1}^{n} \{(l_i, i)\}$.

Computation. The persistence diagram of filtrations of simplicial complexes can be computed with Algorithm 1 below, when $\mathbb{K} = \mathbb{Z}_2$, and when the filtration $\{K_i\}_{1 \leq i \leq n}$ is such that there is a simplex σ_i for which $K_{i+1} = K_i \cup \{\sigma_i\}$ for each i. In Algorithm 1, M is the $n \times n$ matrix such that $m_{ij} = 1 \Leftrightarrow \sigma_i$ is a $(\dim(\sigma_j) - 1)$-face of σ_j, $m_{ij} = 0$ otherwise. M_i is the ith column of M, and l_i is the index of the lowest positive element in M_i, where $l_i = -1$ by convention when $M_i = 0$.

Even though this algorithm has cubic complexity n^3—see [103], faster algorithms can be used in practice depending on the application. For instance, 0-dimensional persistent homology in $\mathbb{Z}_2$ amounts to track the evolution of the connected components, and there exists a very efficient data structure that allows to update the number of connected components in a filtration: the Union-Find data structure (see Chapter I.1. of [69] for more details). This is the data structure we use in practice when we compute 0-dimensional persistent homology.

2.2.4 Stability Properties of Persistence Diagrams

In this section, we introduce the main property of persistence diagrams, i.e. their *stability* with respect to perturbations of their modules. But before presenting the main theorem, we have to detail the metrics we are going to use on the persistence modules and their diagrams. We start with the metric between persistence modules.

The interleaving distance. The so-called *interleaving distance* between persistence modules measures the degree of *interleaving*, i.e. the smallest shift of indices for which one can find commutative diagrams between the modules.

Definition 2.2.10. *Two persistence modules $\mathbb{U}$ and $\mathbb{V}$ are ϵ-interleaved if there exist two families of morphisms $\Psi = \{\psi_\alpha : U_\alpha \to V_{\alpha+\epsilon} : \alpha \in \mathbb{R}\}$ and $\Phi = \{\phi_\alpha : V_\alpha \to U_{\alpha+\epsilon} : \alpha \in \mathbb{R}\}$ such that:*

$$\forall \alpha, \epsilon \in \mathbb{R}, \epsilon > 0, u_{\alpha-\epsilon}^{\alpha+\epsilon} = \phi_\alpha \circ \psi_{\alpha-\epsilon}$$

$$\forall \alpha, \epsilon \in \mathbb{R}, \epsilon > 0, v_{\alpha-\epsilon}^{\alpha+\epsilon} = \psi_\alpha \circ \phi_{\alpha-\epsilon}$$

$$\forall \alpha, \beta, \epsilon \in \mathbb{R}, \alpha \leq \beta, \epsilon > 0, \psi_\beta \circ u_\alpha^\beta = v_{\alpha+\epsilon}^{\beta+\epsilon} \circ \psi_\alpha$$

$$\forall \alpha, \beta, \epsilon \in \mathbb{R}, \alpha \leq \beta, \epsilon > 0, \phi_\beta \circ v_\alpha^\beta = u_{\alpha+\epsilon}^{\beta+\epsilon} \circ \phi_\alpha.$$

It is equivalent to saying that the diagrams in Figure 2.10 commute.

$$\begin{tikzcd}
V_{\alpha-\epsilon} \arrow[rr, "v_{\alpha-\epsilon}^{\alpha+\epsilon}"] \arrow[dr, "\phi_{\alpha-\epsilon}"'] & & V_{\alpha+\epsilon} \\
& U_\alpha \arrow[ur, "\psi_\alpha"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
U_{\alpha-\epsilon} \arrow[rr, "u_{\alpha-\epsilon}^{\alpha+\epsilon}"] \arrow[dr, "\psi_{\alpha-\epsilon}"'] & & U_{\alpha+\epsilon} \\
& V_\alpha \arrow[ur, "\phi_\alpha"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
V_\alpha \arrow[r, "v_\alpha^\beta"] \arrow[dr, "\phi_\alpha"'] & V_\beta \arrow[dr, "\phi_\beta"] & \\
& U_{\alpha+\epsilon} \arrow[r, "u_{\alpha+\epsilon}^{\beta+\epsilon}"'] & U_{\beta+\epsilon}
\end{tikzcd}
\qquad
\begin{tikzcd}
U_\alpha \arrow[r, "u_\alpha^\beta"] \arrow[dr, "\psi_\alpha"'] & U_\beta \arrow[dr, "\psi_\beta"] & \\
& V_{\alpha+\epsilon} \arrow[r, "v_{\alpha+\epsilon}^{\beta+\epsilon}"'] & V_{\beta+\epsilon}
\end{tikzcd}$$

Figure 2.10: Commutative diagrams for interleaving.

Definition 2.2.11. *The interleaving distance* d_{int} *between two persistence modules is defined by:*

$$d_{\text{int}}(\mathbb{U}, \mathbb{V}) = \inf \{\epsilon \geq 0 : \mathbb{U}, \mathbb{V} \textit{ are } \epsilon\textit{-interleaved}\}.$$

The bottleneck distance. We now define the distance between persistence diagrams, called the *bottleneck distance*. The definition of this distance is based on partial matchings between the diagrams. Given two persistence diagrams $\mathrm{Dg}, \mathrm{Dg}'$, a *partial matching* between Dg and Dg' is a subset Γ of $\mathrm{Dg} \times \mathrm{Dg}'$ such that:

$$\forall p \in \mathrm{Dg}, \text{ there is at most one } p' \in \mathrm{Dg}' \text{ such that } (p, p') \in \Gamma,$$

$$\forall p' \in \mathrm{Dg}', \text{ there is at most one } p \in \mathrm{Dg} \text{ such that } (p, p') \in \Gamma.$$

The *cost* of Γ is:

$$\mathrm{cost}(\Gamma) = \max \left\{ \sup_{(p,p') \in \Gamma} \|p - p'\|_\infty, \sup_{p \in \mathrm{Dg} \setminus \Gamma} \|p - \pi_\Delta(p)\|_\infty, \sup_{p' \in \mathrm{Dg}' \setminus \Gamma} \|p' - \pi_\Delta(p')\|_\infty \right\},$$

where, by a slight abuse of notation, we let $\mathrm{Dg} \setminus \Gamma$ denote the set of those points in Dg which have no match in Γ, and similarly for $\mathrm{Dg}' \setminus \Gamma$.

Definition 2.2.12. *Let* $\mathrm{Dg}, \mathrm{Dg}'$ *be two persistence diagrams. The* bottleneck distance *between* Dg *and* Dg' *is:*

$$d_{\text{b}}(\mathrm{Dg}, \mathrm{Dg}') = \inf_\Gamma \ \mathrm{cost}(\Gamma),$$

where Γ *ranges over all partial matchings between* Dg *and* Dg'.

Note that d_{b} is only a pseudometric, not a true metric, because points lying on Δ can be left unmatched at no cost.

The Wasserstein distances. In addition to d_{b}, it is possible to define a 1-parameter family of metrics for persistence diagrams by using the following 1-parameter family of cost functions:

$$\mathrm{cost}_q(\Gamma) = \left(\sum_{(p,p')\in\Gamma} \|p-p'\|_\infty^q + \sum_{p\in \mathrm{Dg}\setminus\Gamma} \|p-\pi_\Delta(p)\|_\infty^q + \sum_{p'\in \mathrm{Dg}'\setminus\Gamma} \|p'-\pi_\Delta(p')\|_\infty^q \right)^{1/q},$$

for any fixed $q \in \mathbb{N}^*$. This is the definition of the *q-Wasserstein distance* $d_{\mathrm{w},q}$:

$$d_{\mathrm{w},q}(\mathrm{Dg}, \mathrm{Dg}') = \inf_{\Gamma} \ \mathrm{cost}_q(\Gamma).$$

Theorem 2.2.13 (Theorem 5.14 in [45])**.** *Let $\mathbb{U}$ and $\mathbb{V}$ be two decomposable persistence modules. Then we have the following inequality:*

$$d_{\mathrm{b}}(\mathrm{Dg}(\mathbb{U}), \mathrm{Dg}(\mathbb{V})) = d_{\mathrm{int}}(\mathbb{U}, \mathbb{V}) \tag{2.1}$$

Note that there exists a similar stability theorem for the Wasserstein distances:

Theorem 2.2.14 (Theorem 3.8 in [113])**.** *Let $\mathbb{U}$ and $\mathbb{V}$ be two decomposable persistence modules. Then we have the following inequality:*

$$d_{\mathrm{w},q}(\mathrm{Dg}(\mathbb{U}), \mathrm{Dg}(\mathbb{V})) \leq (\mathrm{Pers}(\mathbb{U}) + \mathrm{Pers}(\mathbb{V}))^{\frac{1}{q}} d_{\mathrm{int}}(\mathbb{U}, \mathbb{V})^{1-\frac{1}{q}}, \tag{2.2}$$

where $\mathrm{Pers}(\mathbb{U}) = \sum_{p\in\mathrm{Dg}(\mathbb{U})} 2\|p-\pi_\Delta(p)\|_\infty$.

Sublevel sets of functions. An interesting special case is when the persistence modules $\mathbb{U}$ and $\mathbb{V}$ come from the filtrations $\{F_\alpha\}_{\alpha\in\mathbb{R}}$ and $\{G_\alpha\}_{\alpha\in\mathbb{R}}$ induced respectively by tame functions f and g on the same topological space X. Let $\mathrm{Dg}(f)$ and $\mathrm{Dg}(g)$ denote the corresponding persistence diagrams, and let $\|f-g\|_\infty = \sup\{|f(x)-g(x)| \,:\, x\in X\} \leq \epsilon$. Then, $\forall \alpha \in \mathbb{R}$, $F_{\alpha-\epsilon} \subseteq G_\alpha \subseteq F_{\alpha+\epsilon}$ and $G_{\alpha-\epsilon} \subseteq F_\alpha \subseteq G_{\alpha+\epsilon}$. Indeed, let $x \in X$. Then, $f(x) \leq \alpha \ \Rightarrow \ g(x) \leq f(x) + \epsilon \leq \alpha + \epsilon$. The inclusion maps $F_\alpha \hookrightarrow G_{\alpha+\epsilon}$ and $G_\alpha \hookrightarrow F_{\alpha+\epsilon}$ induce an ϵ-interleaving at the homology level. Hence, the interleaving distance is bounded by ϵ, which allows to state the following *stability theorem*:

Theorem 2.2.15. *Let X be a topological space and $f, g : X \to \mathbb{R}$ be tame functions. Then:*

$$d_{\mathrm{b}}(\mathrm{Dg}(f), \mathrm{Dg}(g)) \leq \|f-g\|_\infty \tag{2.3}$$

$$d_{\mathrm{w},q}(\mathrm{Dg}(f), \mathrm{Dg}(g)) \leq (\mathrm{Pers}(f) + \mathrm{Pers}(g))^{\frac{1}{q}} \|f-g\|_\infty^{1-\frac{1}{q}} \tag{2.4}$$

This result is very useful: if one considers a function f and a perturbed version thereof, then the persistence diagrams will be stable in the sense that their bottleneck or Wasserstein distances will be less than the amplitude of the perturbation. Note that Theorem 2.2.15 is significantly weaker for $d_{\mathrm{w},1}$ since, in that case, the upper bound in (2.4) does not depend on $\|f-g\|_\infty$ anymore.

Application on point clouds. An application of Theorem 2.2.15 is the stability of persistence diagrams built from growing balls, as in Figure 1.14. It is stated with the so-called *Hausdorff distance*.

Definition 2.2.16. *Let Y, Z be two subsets of a metric space (X, d_X). Then, the* Hausdorff distance *between Y and Z is:*

$$d_{\mathrm{H}}(Y, Z) = \max\{\sup_{y\in Y} \inf_{z\in Z} d_X(y,z), \sup_{z'\in Z} \inf_{y'\in Y} d_X(z', y')\}. \tag{2.5}$$

Theorem 2.2.17. *Let P, P' be two finite point clouds in a metric space (X, d_X). Let $d_P, d_{P'} : X \to \mathbb{R}_+$ be the distance functions to these point clouds. Then,*

$$d_{\mathrm{b}}(\mathrm{Dg}(d_P), \mathrm{Dg}(d_{P'})) \leq \|d_P - d_{P'}\|_\infty \leq d_{\mathrm{H}}(P, P').$$

2.3 Extended and Levelset Zigzag Persistence

In this section, we present two extensions of persistent homology, called respectively *extended persistence* and *levelset zigzag persistence*. It turns out that they actually encode the same information—see Corollary 2.3.8. However, we use them both in the following chapters of this thesis, since, depending on the type of result we want to prove, the one or the other can be easier to work with. Again, we define these objects for topological spaces, but the definitions hold for simplicial complexes as well.

Notation. From now, on, given a real-valued function f on a topological space X, and an interval $I \subseteq \mathbb{R}$, we denote by X_f^I the preimage $f^{-1}(I)$. We omit the subscript f in the notation when there is no ambiguity in the function considered.

2.3.1 Extended persistence

Filtrations with superlevel sets. Let f be a real-valued function on a topological space X. Recall that the family of sublevel sets of f is nested, and induces a filtration of X. The family $\{X^{[\alpha,+\infty)}\}_{\alpha\in\mathbb{R}}$ of superlevel sets of f is also nested but in the opposite direction: $X^{[\alpha,+\infty)} \supseteq X^{[\beta,+\infty)}$ for all $\alpha \leq \beta \in \mathbb{R}$. We can turn it into a filtration by reversing the real line. Specifically, let $\mathbb{R}^{\mathrm{op}} = \{\tilde{x} : x \in \mathbb{R}\}$, ordered by $\tilde{x} \leq \tilde{y} \Leftrightarrow x \geq y$. We index the family of superlevel sets by $\mathbb{R}^{\mathrm{op}}$, so now we have a filtration: $\{X^{[\tilde{\alpha},+\infty)}\}_{\tilde{\alpha}\in\mathbb{R}^{\mathrm{op}}}$, with $X^{[\tilde{\alpha},+\infty)} \subseteq X^{[\tilde{\beta},+\infty)}$ for all $\tilde{\alpha} \leq \tilde{\beta} \in \mathbb{R}^{\mathrm{op}}$.

Extended filtration. Extended persistence connects the two filtrations at infinity as follows. Replace each superlevel set $X^{[\tilde{\alpha},+\infty)}$ by the pair of spaces $(X, X^{[\tilde{\alpha},+\infty)})$. This maintains the filtration property since we have $(X, X^{[\tilde{\alpha},+\infty)}) \subseteq (X, X^{[\tilde{\beta},+\infty)})$ for all $\tilde{\alpha} \leq \tilde{\beta} \in \mathbb{R}^{\mathrm{op}}$. Then, let $\mathbb{R}_{\mathrm{Ext}} = \mathbb{R} \cup \{+\infty\} \cup \mathbb{R}^{\mathrm{op}}$, where the order is completed by $\alpha < +\infty < \tilde{\beta}$ for all $\alpha \in \mathbb{R}$ and $\tilde{\beta} \in \mathbb{R}^{\mathrm{op}}$. Finally, define the *extended filtration* of f over $\mathbb{R}_{\mathrm{Ext}}$ by:

$$\begin{aligned} F_\alpha &= X^{(-\infty,\alpha]} && \text{for } \alpha \in \mathbb{R} \\ F_{+\infty} &= X \equiv (X, \emptyset) \\ F_{\tilde{\alpha}} &= (X, X^{[\tilde{\alpha},+\infty)}) && \text{for } \tilde{\alpha} \in \mathbb{R}^{\mathrm{op}}, \end{aligned}$$

where we have identified the space X with the pair of spaces $(X, \emptyset)$. This is a well-defined filtration since we have $X^{(-\infty,\alpha]} \subseteq X \equiv (X, \emptyset) \subseteq (X, X^{[\tilde{\beta},+\infty)})$ for all $\alpha \in \mathbb{R}$ and $\tilde{\beta} \in \mathbb{R}^{\mathrm{op}}$. The subfamily $\{F_\alpha\}_{\alpha\in\mathbb{R}}$ is called the *ordinary* part of the filtration, and the subfamily $\{F_{\tilde{\alpha}}\}_{\tilde{\alpha}\in\mathbb{R}^{\mathrm{op}}}$ is called the *relative* part. See Figure 2.11 for an illustration.

Extended persistence module. Inclusions in the extended filtration induce the so-called *extended persistence module* $\mathrm{EP}(f)$:

$$\begin{aligned} \mathrm{EP}(f)_\alpha &= H_*(F_\alpha; \mathbb{K}) = H_*(X^{(-\infty,\alpha]}; \mathbb{K}) && \text{for } \alpha \in \mathbb{R} \\ \mathrm{EP}(f)_{+\infty} &= H_*(F_{+\infty}; \mathbb{K}) = H_*(X; \mathbb{K}) \cong H_*(X, \emptyset; \mathbb{K}) \\ \mathrm{EP}(f)_{\tilde{\alpha}} &= H_*(F_{\tilde{\alpha}}; \mathbb{K}) = H_*(X, X^{[\tilde{\alpha},+\infty)}; \mathbb{K}) && \text{for } \tilde{\alpha} \in \mathbb{R}^{\mathrm{op}}. \end{aligned}$$

Decomposition. As for ordinary persistence, an extended persistence module can be decomposed as a direct sum of interval modules whenever the function f is tame, meaning that the morphisms between the homology groups, either relative or not, of its sub- or superlevel sets have finite rank—see e.g. Section 6.2 in [45]:

$$\mathrm{EP}(f) \simeq \bigoplus_{\alpha\in A} \mathbb{I}[b_\alpha, d_\alpha),$$

where A is an index set, where $b_\alpha \leq d_\alpha \in \mathbb{R}_{\mathrm{Ext}}$, and where each summand $\mathbb{I}[b_\alpha, d_\alpha)$ is made of copies of $\mathbb{K}$ at each index $\beta \in [b_\alpha, d_\alpha)$, and of copies of the zero space elsewhere, the maps between copies of $\mathbb{K}$ being identities.

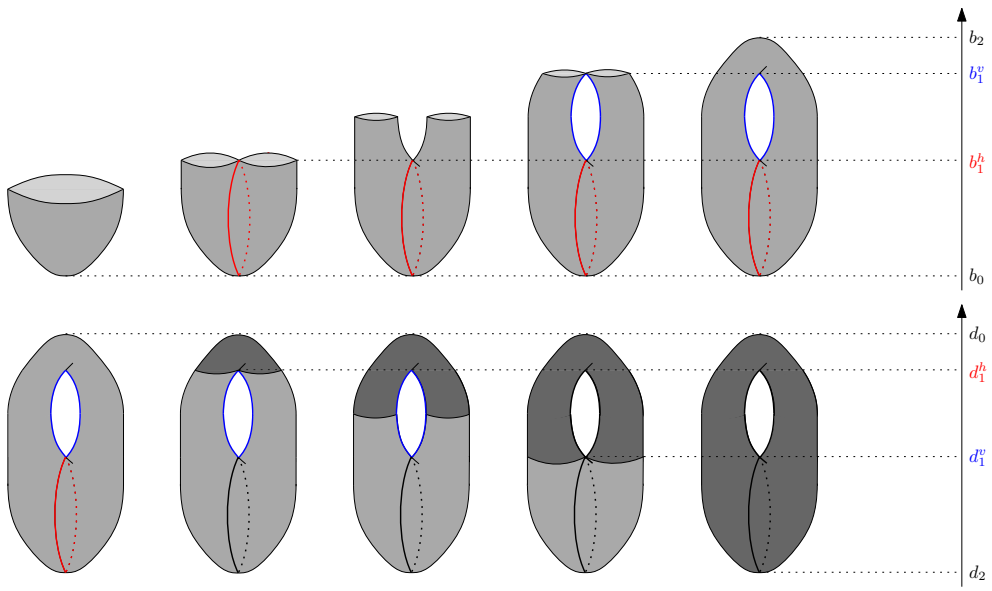

Figure 2.11: The extended filtration of the height function on a torus. The upper row displays the ordinary part of the filtration while the lower row displays the relative part. The red and blue cycles both correspond to extended points in dimension 1. The point corresponding to the red cycle, sometimes called *horizontal cycle*, is located above the diagonal ($d_1^h > b_1^h$), while the point corresponding to the blue cycle, sometimes called *vertical cycle*, is located below the diagonal ($d_1^v > b_1^v$).

Extended persistence diagram. Given a tame function f, its extended persistence module can be encoded in the so-called *extended persistence diagram* $\mathrm{ExDg}(f)$. Moreover, the distinction between ordinary and relative parts of the filtration allows to classify the points in $\mathrm{ExDg}(f)$ in the following way:

- points whose coordinates both belong to $\mathbb{R}$ are called *ordinary* points; they correspond to homological features being born and then dying in the ordinary part of the filtration;
- points whose coordinates both belong to $\mathbb{R}^{\mathrm{op}}$ are called *relative* points; they correspond to homological features being born and then dying in the relative part of the filtration;
- points whose abscissa belongs to $\mathbb{R}$ and whose ordinate belongs to $\mathbb{R}^{\mathrm{op}}$ are called *extended* points; they correspond to homological features being born in the ordinary part and then dying in the relative part of the filtration.

Note that ordinary points lie strictly above the diagonal $\Delta = \{(x, x) : x \in \bar{\mathbb{R}}\}$ and relative points lie strictly below Δ, while extended points can be located anywhere, including on Δ (e.g. for connected components lying inside a single critical level). It is common to decompose $\mathrm{ExDg}(f)$ according to this classification:

$$\mathrm{ExDg}(f) = \mathrm{Ord}(f) \sqcup \mathrm{Rel}(f) \sqcup \mathrm{Ext}^+(f) \sqcup \mathrm{Ext}^-(f),$$

where by convention $\mathrm{Ext}^+(f)$ includes the extended points located on the diagonal Δ.

Persistence measure. From an extended persistence module $\mathrm{EP}(f)$ we derive a measure on the set of rectangles in the plane, called the *persistence measure* and denoted μ_{EP}. Given a rectangle $R = [a, b] \times [c, d]$ with $a < b \leq c < d \in \mathbb{R}_{\mathrm{Ext}}$, we let

$$\mu_{\mathrm{EP}}(R) = r_b^c - r_b^d + r_a^d - r_a^c, \tag{2.6}$$

where r_x^y denotes the rank of the linear map between the vector spaces indexed by $x, y \in \mathbb{R}_{\mathrm{Ext}}$ in $\mathrm{EP}(f)$. When $\mathrm{EP}(f)$ has a well-defined persistence diagram, i.e. f is tame, $\mu_{\mathrm{EP}}(R)$ equals the total multiplicity of the diagram within the rectangle R [45].

Stability. Extended persistence diagrams are also stable in the bottleneck and Wasserstein distances:

Theorem 2.3.1 (EP Stability Theorem in [28]). *For any tame functions $f, g : X \to \mathbb{R}$,*

$$d_{\mathrm{b}}(\mathrm{ExDg}(f), \mathrm{ExDg}(g)) \leq \|f - g\|_\infty \tag{2.7}$$

$$d_{\mathrm{w},q}(\mathrm{ExDg}(f), \mathrm{ExDg}(g)) \leq (\mathrm{Pers}(f) + \mathrm{Pers}(g))^{\frac{1}{q}} \|f - g\|_\infty^{1-\frac{1}{q}} \tag{2.8}$$

Moreover, as pointed out in [55], the theorem can be strengthened to apply to each subdiagram Ord, Ext^+, Ext^-, Rel and to each homological dimension individually.

Extended persistence diagrams also enjoy a *symmetry theorem* when X is a d-manifold, $d \in \mathbb{N}^*$.

Theorem 2.3.2 (Symmetry Theorem in [55]). *Let $R : (x, y) \mapsto (-x, -y)$. Then, for any tame function $f : X \to \mathbb{R}$ defined on a d-manifold X, one has, for any homological dimension $p < d$:*
(i) $\mathrm{Ord}_p(f) = R(\mathrm{Ord}_{d-p-1}(-f))$ (ii) $\mathrm{Rel}_p(f) = R(\mathrm{Rel}_{d-p-1}(-f))$ (iii) $\mathrm{Ext}_p(f) = R(\mathrm{Ext}_{d-p}(-f))$

2.3.2 Levelset zigzag persistence

Levelset zigzag persistence [28] is specifically designed for the so-called *Morse-type functions* and the stratification of the space they induce.

Morse-type functions. Morse-type functions are generalizations of the classical Morse functions that share some of their properties without having to be differentiable nor defined over a smooth manifold.

Definition 2.3.3 (Morse-type [28]). *A continuous real-valued function f on a topological space X is* of Morse type *if:*

(i) *There is a finite set* $\mathrm{Crit}(f) = \{a_1 < ... < a_n\}$, *called the set of* critical values, *such that for every open interval $(a_0 = -\infty, a_1), ..., (a_i, a_{i+1}), ..., (a_n, a_{n+1} = +\infty)$ there is a compact and locally connected space Y_i and a homeomorphism $\mu_i : Y_i \times (a_i, a_{i+1}) \to X^{(a_i, a_{i+1})}$ such that $\forall i = 0, ..., n, f|_{X^{(a_i, a_{i+1})}} = \pi_2 \circ \mu_i^{-1}$, where π_2 is the projection onto the second factor;*

(ii) *$\forall i = 1, ..., n-1, \mu_i$ extends to a continuous function $\bar{\mu}_i : Y_i \times [a_i, a_{i+1}] \to X^{[a_i, a_{i+1}]}$ – similarly μ_0 extends to $\bar{\mu}_0 : Y_0 \times (-\infty, a_1] \to X^{(-\infty, a_1]}$ and μ_n extends to $\bar{\mu}_n : Y_n \times [a_n, +\infty) \to X^{[a_n, +\infty)}$;*

(iii) *Each levelset X^t has a finitely-generated homology.*

Items (i) and (ii) define a stratification of X, which we use extensively in the next chapters. Morse functions are known to be of Morse type while the converse is clearly not true. Moreover, it follows from item (iii) in Definition 2.3.3 that Morse-type functions induce *pointwise finite-dimensional* (pfd) extended persistence modules, meaning that each vector space in the extended persistence module has finite dimension. Clearly, pfd persistence modules are tame and thus decomposable.

Zigzag persistence modules. A zigzag persistence module is a discrete persistence module whose arrows can go indifferently forward *or* backward.

Definition 2.3.4. *Let $\mathbb{K}$ be a field and $n \in \mathbb{N}$. A* zigzag persistence module *$\mathbb{U}$ is a set of $\mathbb{K}$-vector spaces indexed by $\{1, \cdots, n\}$, denoted by $\{U_i\}_{1 \leq i \leq n}$, together with a family of linear maps $\{u_i : U_i \leftrightarrow U_{i+1}\}_{1 \leq i \leq n-1}$, where $\leftrightarrow$ means that the linear map is either $\to$ or $\leftarrow$.*

As for usual persistence modules, any sequence of topological spaces with canonical inclusions (going either forward or backward) induces a zigzag persistence module after computing the homology groups. Note however that the full sequence is not required to be a filtration anymore. Particular zigzag persistence modules, called *levelset zigzag persistence modules*, can be defined for Morse-type functions.

Definition 2.3.5. *Let $f : X \to \mathbb{R}$ be a Morse-type function, and let* $\mathrm{Crit}(f) = \{a_1, \cdots, a_n\}$ *be its set of critical values. Let $-\infty = a_0 < s_0 < a_1 < s_1 < a_2 < \cdots < s_{n-1} < a_n < s_n < a_{n+1} = +\infty$. Then, for any $1 \leq i \leq j \leq n$, we write X_i^j for $X^{[s_i,s_j]}$, and we define the* levelset zigzag *as the following sequence of $2n+1$ nodes:*

$$X_0^0 \hookrightarrow X_0^1 \hookleftarrow X_1^1 \hookrightarrow X_1^2 \hookleftarrow \cdots \hookrightarrow X_{n-1}^n \hookleftarrow X_n^n,$$

where each arrow is an inclusion. Computing the homology groups of each space and the linear maps induced from the corresponding inclusions gives the so-called levelset zigzag persistence module $\mathrm{LZZ}(f)$*:*

$$H_*(X_0^0;\mathbb{K}) \to H_*(X_0^1;\mathbb{K}) \leftarrow H_*(X_1^1;\mathbb{K}) \to H_*(X_1^2;\mathbb{K}) \leftarrow \cdots \to H_*(X_{n-1}^n;\mathbb{K}) \leftarrow H_*(X_n^n;\mathbb{K}).$$

Decomposition. Zigzag persistence modules also enjoy a decomposition theorem:

Theorem 2.3.6 (Theorem 2.5 in [27])**.** *Any zigzag persistence module $\mathbb{U}$ decomposes as a direct sum of closed interval modules:*

$$\mathbb{U} \simeq \bigoplus_{\alpha \in A} \mathbb{I}_{[b_\alpha, d_\alpha]},$$

where A is an index set, where $1 \leq b_\alpha \leq d_\alpha \leq n$, and where each summand $\mathbb{I}_{[b_\alpha,d_\alpha]}$ is made of copies of $\mathbb{K}$ at each index $\beta \in [b_\alpha, d_\alpha]$, and of copies of the zero space elsewhere, the maps between copies of $\mathbb{K}$ being identities.

In the case of a levelset zigzag persistence module induced by a Morse-type function f, each closed interval $[b_\alpha, d_\alpha]$ is of the form $[X_{i_\alpha}^{i'_\alpha}, X_{j_\alpha}^{j'_\alpha}]$, where i'_α is either i_α or $i_\alpha + 1$ and similarly for j'_α, hence the following classification:

Type	I	II	III	IV
	$i'_\alpha = i_\alpha + 1$ $j'_\alpha = j_\alpha$	$i'_\alpha = i_\alpha$ $j'_\alpha = j_\alpha + 1$	$i'_\alpha = i_\alpha + 1$ $j'_\alpha = j_\alpha + 1$	$i'_\alpha = i_\alpha$ $j'_\alpha = j_\alpha$

The disjoint union of all of these intervals is called the *levelset zigzag persistence barcode* $\mathrm{LBc}(f)$.

Mayer-Vietoris half-pyramid. Morse-type functions also allow us to build the so-called *Mayer-Vietoris half-pyramid* [28], which is the diagram of topological spaces and inclusions displayed in Figure 2.12. Any zigzag within the Mayer-Vietoris half-pyramid that stretches from the left boundary (i.e. the node X_0^0) to the right boundary without backtracking is called *monotone*. Theorem 2.3.7 and Corollary 2.3.8 below allow us to link the zigzag persistence modules of any pair of monotone zigzags in the half-pyramid. We use these results extensively in Section 4.3.1.

Theorem 2.3.7 (Pyramid Theorem in [28])**.** *For any Morse-type function f, there exists a bijection between the barcodes of any pair of monotone zigzag persistence modules in the Mayer-Vietoris half-pyramid.*

Since the extended persistence module of f is a monotone zigzag persistence module—more precisely the principal diagonal—of the Mayer-Vietoris half-pyramid, we have the following corollary.

Corollary 2.3.8 (Table 1 in [28])**.** *For any Morse-type function f, there exists a bijection between* $\mathrm{ExDg}(f)$ *and* $\mathrm{LBc}(f)$*, which is described in Table 2.1.*

In particular, bottleneck and Wasserstein distances, as well as stability results, can be derived for levelset zigzag persistence barcodes using this correspondence and Theorem 2.3.1.

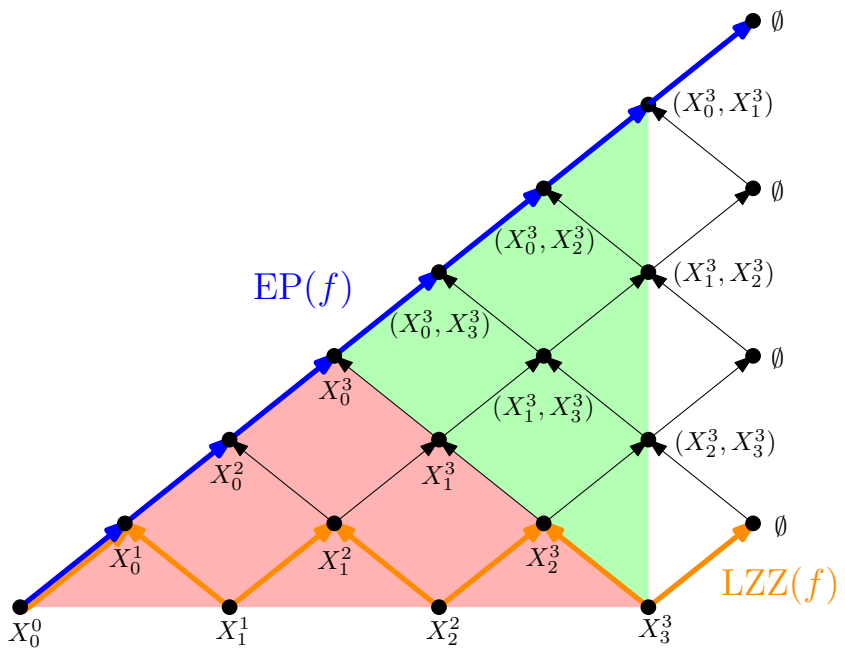

Figure 2.12: We show the Mayer-Vietoris half-pyramid when the Morse-type function has 3 critical values. It is composed of two faces of the full Mayer-Vietoris pyramid: the south face (red) and the east face (green). The extended persistence module EP(f) is in blue and the levelset zigzag persistence module LZZ(f) is in orange.

Type	Ord	Rel	Ext$^+$	Ext$^-$
ExDg(f)	$[a_i, a_j)$	$[\tilde{a}_j, \tilde{a}_i)$	$[a_i, \tilde{a}_j)$	$[a_j, \tilde{a}_i)$
LBc(f)	$[X_{i-1}^i, X_{j-1}^{j-1}]$	$[X_i^i, X_{j-1}^j]^-$	$[X_{i-1}^i, X_{j-1}^j]$	$[X_i^i, X_{j-1}^{j-1}]^-$
Type	I	II	III	IV

Table 2.1: This table gives the correspondences between the points of ExDg(f) and the intervals of LBc(f). The minus sign on some intervals of LBc(f) means that the homological dimension of that interval is equal to the dimension of its corresponding point in ExDg(f) minus 1.

2.4 Reeb graphs

The Reeb graph provides a meaningful alternative to persistence diagrams as summary of a topological space and a real-valued function defined on that space. Intuitively, it continuously collapses the connected components of the level sets of the function into single points, thus tracking the values of the functions at which the connected components merge or split. Reeb graphs have been widely used in computer graphics and visualization—see [12] for a survey.

Definition 2.4.1. *Given a topological space X and a continuous function $f : X \to \mathbb{R}$, we define the equivalence relation $\sim_f$ between points of X by:*

$$x \sim_f y \;\Leftrightarrow\; \left[f(x) = f(y) \text{ and } x, y \text{ belong to the same connected component of } f^{-1}(f(x)) = f^{-1}(f(y))\right].$$

The Reeb graph $\mathrm{R}_f(X)$ *is the quotient space $X/\sim_f$.*

As f is constant on equivalence classes, there is an induced map $\tilde{f} : \mathrm{R}_f(X) \to \mathbb{R}$ such that

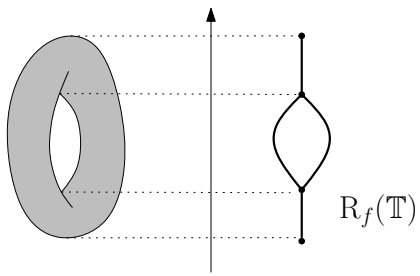

Figure 2.13: Reeb graph of the height function f of an embedding of the torus $\mathbb{T}$ in $\mathbb{R}^3$. Note how the critical points induce changes on the graph.

$f = \tilde{f} \circ \pi$, where π is the quotient map $X \to \mathrm{R}_f(X)$:

$$\begin{tikzcd} X \arrow[rr, "\pi"] \arrow[dr, "f"'] & & \mathrm{R}_f(X) \arrow[dl, "\tilde{f}"] \\ & \mathbb{R} & \end{tikzcd} \tag{2.9}$$

If f is a function of Morse type, then the pair (X, f) is an $\mathbb{R}$-*constructible space* in the sense of [61]. This ensures that the Reeb graph is a multigraph, whose nodes are in one-to-one correspondence with the connected components of the critical level sets of f. In that case, computing the Reeb graph of a Reeb graph preserves all information, as stated in the following remark.

Remark 2.4.2. *Let f be a Morse-type function. Then there is a bijection $b : \mathrm{R}_{\tilde{f}}(\mathrm{R}_f(X)) \to \mathrm{R}_f(X)$ such that $\tilde{f} \circ b = \tilde{\tilde{f}}$. In other words, computing the Reeb graph is an idempotent operation.*

Reeb graphs as metric spaces. Any Reeb graph can be turned into a metric space by adequately defining a metric between any pair of its points.

Definition 2.4.3 ([8])**.** *Let X be a topological space and $f : X \to \mathbb{R}$ be a continuous function. Then, we define the following metric between any pair $x, x' \in \mathrm{R}_f(X)$:*

$$d_f(x, x') = \min_{\pi : x \to x'} \left\{ \max_{t \in [0,1]} \tilde{f} \circ \pi(t) - \min_{t \in [0,1]} \tilde{f} \circ \pi(t) \right\},$$

where $\pi : [0, 1] \to \mathrm{R}_f(X)$ ranges over the continuous paths from x to x' in $\mathrm{R}_f(X)$ with $\pi(0) = x$ and $\pi(1) = x'$.

2.4.1 Persistence-based bag-of-features signature

There is a nice interpretation of $\mathrm{ExDg}(\tilde{f})$ in terms of the structure of $\mathrm{R}_f(X)$. We refer the reader to [8] and the references therein for a full description as well as formal definitions and statements. Orienting the Reeb graph vertically so $\tilde{f}$ is the height function, we can see each connected component of the graph as a trunk with multiple branches (some oriented upwards, others oriented downwards) and holes. Then, one has the following correspondences, where the *vertical span* of a feature is the span of its image under $\tilde{f}$:

- The vertical spans of the trunks are given by the points in $\mathrm{Ext}_0^+(\tilde{f})$;
- The vertical spans of the branches that are oriented downwards are given by the points in $\mathrm{Ord}_0(\tilde{f})$;
- The vertical spans of the branches that are oriented upwards are given by the points in $\mathrm{Rel}_1(\tilde{f})$;
- The vertical spans of the holes are given by the points in $\mathrm{Ext}_1^-(\tilde{f})$.

The rest of the diagram of $\tilde{f}$ is empty. These correspondences provide a dictionary to read off the structure of the Reeb graph from the persistence diagram of the induced map $\tilde{f}$. Note that it is a bag-of-features type signature, taking an inventory of all the features (trunks, branches, holes) together with their vertical spans, but leaving aside the actual layout of the features. As a consequence, it is an incomplete signature: two Reeb graphs with the same persistence diagram may not be isomorphic, as illustrated in Figure 2.14.

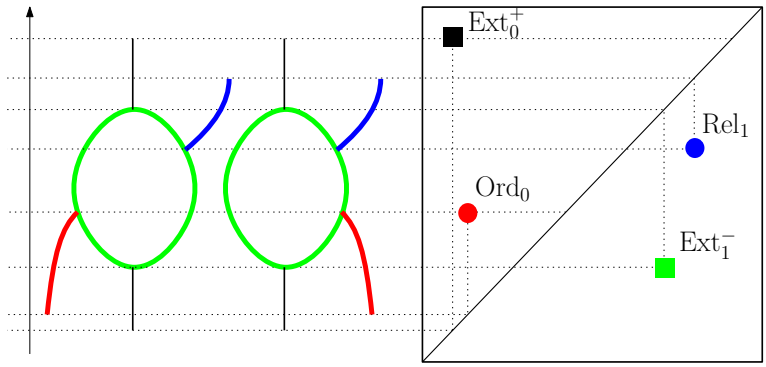

Figure 2.14: Two Reeb graphs with the same set of features but not the same layout.

Connection to the extended and levelset zigzag persistence of f.

We now show that the topological structure of $\mathrm{R}_f(X)$ is actually nothing but a simplification of the one of f, which we phrase in terms of persistence diagrams. This result and its proof can be seen as an exercise combining all concepts introduced before in a simple way.

Theorem 2.4.4. *Let X be a topological space and $f : X \to \mathbb{R}$ be a function of Morse type. Then, the levelset zigzag persistence barcodes of f and $\tilde{f}$ in dimension 0 are the same: $\mathrm{LBc}_0(f) = \mathrm{LBc}_0(\tilde{f})$, and the extended persistence diagram of $\tilde{f}$ is included in the one of f: $\mathrm{ExDg}(\tilde{f}) \subseteq \mathrm{ExDg}(f)$. More precisely:*

$$\begin{aligned}
\mathrm{ExDg}_0(\tilde{f}) &= \mathrm{ExDg}_0(f) \\
\mathrm{ExDg}_1(\tilde{f}) &= \mathrm{ExDg}_1(f) \setminus (\mathrm{Ext}_1^+(f) \cup \mathrm{Ord}_1(f)) \\
\mathrm{ExDg}_p(\tilde{f}) &= \emptyset \text{ if } p \geq 2
\end{aligned}$$

Note that $\mathrm{Ext}_0^-(\tilde{f}) = \emptyset$ because every essential 0-dimensional feature corresponds to some connected component of the domain, and it is born at the minimum function value and killed at the maximum function value over that connected component, hence it belongs to Ext_0^+. Similarly, $\mathrm{Rel}_0(\tilde{f}) = \emptyset$ because no 0-dimensional homology class (i.e. connected component) can be created in the relative part of the extended filtration of f. Hence, the structure of a Reeb graph can be read off from the levelset zigzag persistence module of $\tilde{f}$. Indeed, since $\mathrm{Ext}_1^+(\tilde{f})$, $\mathrm{Ord}_1(\tilde{f}), \mathrm{Ext}_0^-(\tilde{f}), \mathrm{Rel}_0(\tilde{f})$ and $\mathrm{ExDg}_p(\tilde{f})$ for $p \geq 2$ are empty, it follows from Corollary 2.3.8 that there is a bijection preserving types between $\mathrm{ExDg}_0(\tilde{f}) \cup \mathrm{ExDg}_1(\tilde{f})$ and $\mathrm{LBc}_0(\tilde{f})$. This is because all intervals in the 1-dimensional extended persistence module of $\tilde{f}$ are either of type Rel or Ext^-, and thus their analogues in the levelset zigzag persistence module of $\tilde{f}$ have homological dimension 0 according to Table 2.1.

We now provide a proof of Theorem 2.4.4 for completeness, as we have not seen this result stated formally in the literature. First, note that $\mathrm{Crit}(f) = \{a_1, \cdots, a_n\} = \mathrm{Crit}(\tilde{f})$. Hence, given $i \leq j$ and $[s_i, s_j]$ as in Definition 2.3.5, we recall that X_i^j denote $X^{[s_i,s_j]} = f^{-1}([s_i, s_j])$ and $\mathrm{R}_f(X)_i^j$ denote $\mathrm{R}_f(X)^{[s_i,s_j]} = \tilde{f}^{-1}([s_i, s_j])$.

Lemma 2.4.5. *Let π denote the quotient map $X \to \mathrm{R}_f(X)$. Let $i \leq j$. Then the morphism $\pi_* : H_0(X_i^j) \to H_0(\mathrm{R}_f(X)_i^j)$ is an isomorphism.*

The proof of Lemma 2.4.5 is simpler when π admits *continuous sections*, i.e. when there exist continuous maps $\sigma : \mathrm{R}_f(X) \to X$ such that $\pi \circ \sigma = \mathrm{id}_{\mathrm{R}_f(X)}$. Below we give the proof under this

hypothesis, deferring the general case of Morse-type functions to Appendix A. The hypothesis holds for instance when X is a compact smooth manifold and f is a Morse function, or when X is a simplicial complex and f is piecewise-linear.

Proof. Since π is surjective, proving the result boils down to showing that x, y are connected in X_i^j if and only if $\pi(x), \pi(y)$ are connected in $\mathrm{R}_f(X)_i^j$.

- If x, y are connected in X_i^j, then $\pi(x), \pi(y)$ are connected in $\mathrm{R}_f(X)_i^j$ by continuity of π and commutativity of (2.9).
- If $\pi(x), \pi(y)$ are connected in $\mathrm{R}_f(X)_i^j$, then choose a path γ connecting $\pi(x)$ to $\pi(y)$. By definition of σ, we have $\pi \circ \sigma \circ \pi(x) = \pi(x)$, thus $\sigma \circ \pi(x)$ and x lie in the same connected component of $f^{-1}(f(x))$. Let γ_x be a path connecting x to $\sigma \circ \pi(x)$. Similarly, let γ_y be a path connecting $\sigma \circ \pi(y)$ to y. Then, $\gamma_y \circ \sigma(\gamma) \circ \gamma_x$ is a path between x and y in X_i^j.

□

Proof of Theorem 2.4.4. We first show that $\mathrm{LBc}_0(f) = \mathrm{LBc}_0(\tilde{f})$. Let π denote the quotient map $X \to \mathrm{R}_f(X)$. Since π is continuous, it induces a morphism in homology π_*. We will show that π_* induces an isomorphism between $\mathrm{LZZ}(f)$ and $\mathrm{LZZ}(\tilde{f})$ in dimension 0.

Now, let $1 \leq i \leq n$.

- According to Lemma 2.4.5, $\pi_* : H_0(X_i^i) \to H_0(\mathrm{R}_f(X)_i^i)$ is an isomorphism, and the same holds for $\pi_* : H_0(X_i^{i+1}) \to H_0(\mathrm{R}_f(X)_i^{i+1})$. Hence π_* induces a pointwise isomorphism in dimension 0 between $\mathrm{LZZ}(f)$ and $\mathrm{LZZ}(\tilde{f})$. Since $\mathrm{Crit}(f) = \{a_1, \cdots, a_n\} = \mathrm{Crit}(\tilde{f})$, it follows that both $\mathrm{LZZ}(f)$ and $\mathrm{LZZ}(\tilde{f})$ have $2n+1$ nodes.
- Let $\iota : X_i^i \to X_i^{i+1}$ and $\iota^{\mathrm{R}} : \mathrm{R}_f(X)_i^i \to \mathrm{R}_f(X)_i^{i+1}$ be canonical inclusions. Then, we have $\pi \circ \iota = \iota^{\mathrm{R}} \circ \pi$ by definition of ι^{R}. Hence, the following diagram commutes:

$$\begin{tikzcd} H_0(X_i^i) \arrow[r, "\iota_*"] \arrow[d, "\pi_*"'] & H_0(X_i^{i+1}) \arrow[d, "\pi_*"] \\ H_0(\mathrm{R}_f(X)_i^i) \arrow[r, "\iota_*^{\mathrm{R}}"'] & H_0(\mathrm{R}_f(X)_i^{i+1}) \end{tikzcd}$$

  and the same is true for the canonical inclusions $X_{i-1}^i \hookleftarrow X_i^i$ and $\mathrm{R}_f(X)_{i-1}^i \hookleftarrow \mathrm{R}_f(X)_i^i$.

Hence, the induced pointwise isomorphism is an isomorphism between $\mathrm{LZZ}_0(f)$ and $\mathrm{LZZ}_0(\tilde{f})$.

Now, recall that there is a bijection b_1 preserving types between $\mathrm{ExDg}(\tilde{f})$ and $\mathrm{LBc}_0(\tilde{f})$. Since there is also a bijection b_2 preserving types between $\mathrm{LBc}_0(\tilde{f})$ and $\mathrm{LBc}_0(f)$ and a bijection b_3 preserving types between $\mathrm{LBc}_0(f)$ and $\mathrm{Ord}_0(f) \cup \mathrm{Ext}_0^+(f) \cup \mathrm{Rel}_1(f) \cup \mathrm{Ext}_1^-(f)$ from Corollary 2.3.8, the result follows by considering the bijection $b_3 \circ b_2 \circ b_1$. □

2.4.2 Metrics between Reeb graphs

Finding relevant dissimilarity measures for comparing Reeb graphs has become an important question in the recent years. The quality of a dissimilarity measure is usually assessed through three criteria: its ability to satisfy the axioms of a metric, its discriminative power, and its computational efficiency. The most natural choice to begin with is to use the *Gromov-Hausdorff distance* d_{GH} [22] for Reeb graphs seen as metric spaces—see Definition 2.4.3. The main drawback of this distance is to quickly become intractable to compute in practice, even for graphs that are metric trees [2]. Among recent contributions, the *functional distortion distance* d_{FD} [8], the *interleaving distance* d_{I} [61]—which is equivalent to d_{FD} [10]— and the *edit distance* d_{E} [7, 66] share the same advantages and drawbacks as d_{GH}, in particular they enjoy good stability and discriminativity properties but they lack efficient algorithms for their computation, moreover they can be difficult to interpret. By contrast, the *bottleneck distance* d_{b} compares Reeb graphs with their extended persistence diagrams—which act as stable bag-of-features signatures—and can be computed efficiently in practice. Its main drawback though is to be only a pseudometric, so distinct graphs can have the same signature and therefore be deemed equal in d_{b}—see Figure 2.14. We now give details on these distances.

The Gromov-Hausdorff distance d_{GH}. This distance compares Reeb graphs by computing the length distortion of corresponding curves drawn on the graphs.

Definition 2.4.6. *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous functions. The* Gromov-Hausdorff distance *between $\mathrm{R}_f(X)$ and $\mathrm{R}_g(Y)$ is:*

$$d_{\mathrm{GH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) = \inf_{\phi,\psi} D(\phi, \psi), \tag{2.10}$$

where:

- *$\phi : \mathrm{R}_f(X) \to \mathrm{R}_g(Y)$ and $\psi : \mathrm{R}_g(Y) \to \mathrm{R}_f(X)$ are (nonnecessarily continuous) maps,*
- *$D(\phi, \psi) = \frac{1}{2} \sup \left\{|d_f(x, x') - d_g(y, y')| : (x, y), (x', y') \in C(\phi, \psi)\right\}$,*
- *$C(\phi, \psi) = \{(x, \phi(x)) : x \in \mathrm{R}_f(X)\} \cup \{(\psi(y), y) : y \in \mathrm{R}_g(Y)\}$.*

The main drawback of d_{GH} is that it does not fully take function values into account. For instance, it is straightforward to show that $d_{\mathrm{GH}}(\mathrm{R}_f(X), \mathrm{R}_{-f}(X)) = 0$ and $d_{\mathrm{GH}}(\mathrm{R}_f(X), \mathrm{R}_{f+c}(X)) = 0$, where $c \in \mathbb{R}$.

Functional distances. To handle this issue, Bauer et al. [9] suggested to add terms corresponding to the absolute difference in function values:

Definition 2.4.7 ([9])**.** *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous functions. The* functional Gromov-Hausdorff distance *between $\mathrm{R}_f(X)$ and $\mathrm{R}_g(Y)$ is:*

$$d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) = \inf_{\phi,\psi} \max\{D(\phi, \psi), \|f - g \circ \phi\|_\infty, \|f \circ \psi - g\|_\infty\}, \tag{2.11}$$

where ϕ, ψ and $D(\phi, \psi)$ are as in Definition 2.4.6.

In Section 3.2, we will show that the functional Gromov-Hausdorff distance is actually *locally* equivalent to the bottleneck distance between the extended persistence diagrams of the functions. To ease the analysis, we will use a third distance which constrains the maps ϕ and ψ to be continuous.

Definition 2.4.8 ([8])**.** *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous functions. The* functional distortion distance *between $\mathrm{R}_f(X)$ and $\mathrm{R}_g(Y)$ is:*

$$d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) = \inf_{\phi,\psi} \max\{D(\phi, \psi), \|f - g \circ \phi\|_\infty, \|f \circ \psi - g\|_\infty\}, \tag{2.12}$$

where ϕ, ψ and $D(\phi, \psi)$ are as in Definition 2.4.6, and where we also require ϕ and ψ to be continuous.

Requiring the maps to be continuous has very little impact on the distance properties since d_{fGH} and d_{FD} are *strongly equivalent*:

Theorem 2.4.9 (Theorem 5.1 in [9])**.** *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous functions. Then:*

$$d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq 3 d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)).$$

Furthermore, these distances are stable with respect to changes in the function:

Theorem 2.4.10 (Theorem 4.1 in [8])**.** *Let X be a topological space and let $f, g : X \to \mathbb{R}$ be two Morse-type functions with continuous sections. Then:*

$$d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(X)) \leq d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(X)) \leq \|f - g\|_\infty.$$

The bottleneck distance d_{b}. This distance uses the extended persistence diagrams of the functions to compare the Reeb graphs.

Definition 2.4.11. *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous and tame functions. The* bottleneck distance *between $\mathrm{R}_f(X)$ and $\mathrm{R}_g(Y)$ is:*

$$d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) = d_{\mathrm{b}}(\mathrm{ExDg}(\tilde{f}), \mathrm{ExDg}(\tilde{g})), \tag{2.13}$$

where $\tilde{f} : \mathrm{R}_f(X) \to \mathbb{R}$ and $\tilde{g} : \mathrm{R}_g(Y) \to \mathbb{R}$ are the induced maps on the Reeb graphs.

As a direct application of the stability theorem—see Theorem 2.3.1, the bottleneck distance is also stable with respect to changes in the function.

A first inequality. Bauer et al. [8] related d_{FD} and d_{b} as follows:

Theorem 2.4.12 (Theorem 4.3 in [8])**.** *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous and tame functions. The following inequality holds:*

$$d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq 3\, d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)).$$

This result can be improved using the end of Section 3.4 of [15] (using the fact that levelset zigzag persistence barcodes and extended persistence diagrams are essentially the same—see Corollary 2.3.8), and then Lemma 9 of [10] and Theorem 2.4.9:

Theorem 2.4.13. *Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be continuous and tame functions. The following inequality holds:*

$$d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq 2\, d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq 6\, d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)).$$

2.4.3 Simplification techniques

Being able to simplify Reeb graphs by removing small topological features is very useful. It very often helps to prove theoretical results concerning Reeb graphs, and has many applications, for instance in Reeb graph computation and visualization [74, 115, 134].

In this section, we define one possible way to do such a simplification, that we will use in Section 3.2 of Chapter 3 to prove Theorem 3.0.1. We recall that, due to the bag-of-feature interpretation of $\mathrm{ExDg}(\tilde{f})$, any point $p = (a, b) \in \mathrm{ExDg}(\tilde{f}) \setminus \mathrm{Ext}_0^+(\tilde{f})$ represents either an upward branch, a downward branch or a loop. Depending on the feature type, we define the *merging path* π^p as follows:

- assume $p \in \mathrm{Ext}_1^-(\tilde{f})$, i.e. p represents a loop with extremities $x_1, x_2 \in \mathrm{R}_f(X)$, so that we have $\tilde{f}(x_1) = a$ and $\tilde{f}(x_2) = b$. Let π_1^p and π_2^p be two disjoint sub-curves of the loop that connect x_1 and x_2. Then, we let $\pi^p = \pi_1^p \cup \pi_2^p$.
- assume $p \in \mathrm{Ord}_0(\tilde{f}) \cup \mathrm{Rel}_1(\tilde{f})$, i.e. p represents a branch. Let C_1 be this branch. If $p \in \mathrm{Ord}_0(\tilde{f})$ (resp. $\mathrm{Rel}_1(\tilde{f})$), let C_2 be an arbitrary connected component of $\tilde{f}^{-1}((-\infty, b))$ (resp. $\tilde{f}^{-1}((b, +\infty))$) to which C_1 gets connected at level b. We define the triple x_1, x_2, y as follows:

  $$\begin{cases} x_1 = \mathrm{argmin}_{x \in C_1} \tilde{f}(x), y = \mathrm{argmax}_{x \in C_1} \tilde{f}(x) \text{ and } x_2 = \mathrm{argmin}_{x \in C_2} \tilde{f}(x) \text{ if } p \in \mathrm{Ord}_0(\tilde{f}) \text{ and} \\ x_1 = \mathrm{argmax}_{x \in C_1} \tilde{f}(x), y = \mathrm{argmin}_{x \in C_1} \tilde{f}(x) \text{ and } x_2 = \mathrm{argmax}_{x \in C_2} \tilde{f}(x) \text{ if } p \in \mathrm{Rel}_1(\tilde{f}). \end{cases}$$

  Note that we have $\tilde{f}(x_1) = a$ and $\tilde{f}(y) = b$ in both cases. We now let π_1^p be any arbitrary path from x_1 to y and π_2^p be any arbitrary path from y to x_2. Finally, we let $\pi^p = \pi_1^p \cup \pi_2^p$ as before.

Definition 2.4.14 ([8, 74, 115])**.** *Let X be a topological space and $f : X \to \mathbb{R}$ a Morse-type function. Let $p \in \mathrm{ExDg}(\tilde{f}) \setminus \mathrm{Ext}_0^+(\tilde{f})$. We define the equivalence relation $\sim_p$ as follows:*

$$x \sim_p x' \Leftrightarrow x, x' \in \pi^p \text{ and } \tilde{f}(x) = \tilde{f}(x').$$

See Figure 2.15 for an illustration.

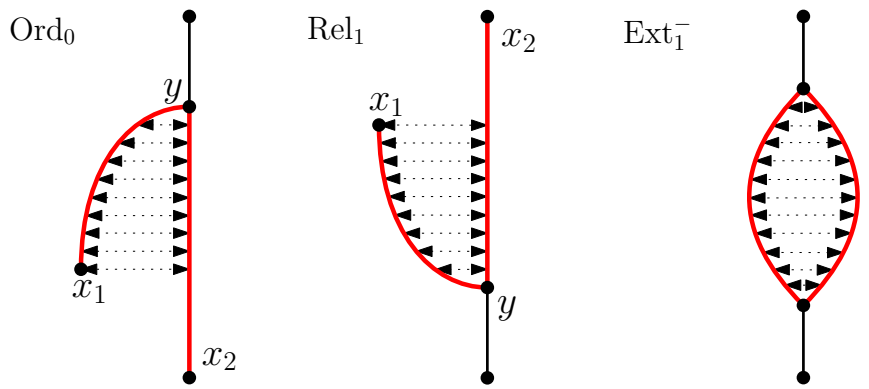

Figure 2.15: Dotted arrows show points that are glued together by the simplification operator depending on the topological feature p, whose merging path π^p is highlighted in red.

Definition 2.4.15. *Let X be a topological space and $f : X \to \mathbb{R}$ a Morse-type function. Let $\alpha > 0$, $\mathrm{Feat}_\alpha = \{p \in \mathrm{ExDg}(\tilde{f}) \setminus \mathrm{Ext}_0^+(\tilde{f}) : 2d_\infty(p, \Delta) \leq \alpha\}$ the set of points of $\mathrm{ExDg}(\tilde{f})$ representing loops and branches of $\mathrm{R}_f(X)$ whose vertical span is less than α and Conn_α the set of connected components of $\mathrm{R}_f(X)$ whose vertical span is less than α. Finally, let $\sim_\alpha$ be the transitive closure of all $\sim_p$, where $p \in \mathrm{Feat}_\alpha$. The* simplification operator *S_α is defined as:*

$$S_\alpha(\mathrm{R}_f(X)) = (\mathrm{R}_f(X) \setminus \mathrm{Conn}_\alpha)/\sim_\alpha .$$

An illustration of the action of this operator is shown in the left part of Figure 3.7. Intuitively, the simplification operator S_α removes all features whose vertical span is less than α (in an arbitrary order) without perturbing the other features too much. We state this property in the following Lemma:

Lemma 2.4.16 (Theorem 7.3 and following remark in [9])**.** *Given $\alpha > 0$, the simplification operator S_α takes any Reeb graph R_h to $\mathrm{R}_{h'} = S_\alpha(\mathrm{R}_h)$ such that $\mathrm{ExDg}(h') \cap \mathrm{off}_{\alpha/2}(\Delta) = \emptyset$ and*

$$d_{\mathrm{b}}(\mathrm{R}_h, \mathrm{R}_{h'}) \leq 2\, d_{\mathrm{FD}}(\mathrm{R}_h, \mathrm{R}_{h'}) \leq 4\alpha,$$

where $\mathrm{off}_{\alpha/2}(\Delta) = \{x \in \mathbb{R}^2 : d_\infty(x, \Delta) \leq \alpha/2\}$ is the $(\alpha/2)$-offset of the diagonal Δ in the ℓ_∞-distance.

2.4.4 Computation

One issue with the Reeb graph is the computation of the graph itself. Indeed, when the pair (X, f) is known only through a finite set of measurements, the graph can only be approximated within a certain error. Building approximations from finite point samples with scalar values is a problem in its own right. A natural approach is to build a simplicial complex on top of the point samples, to serve as a proxy for the underlying continuous space; then, to extend the scalar values at the vertices to a piecewise-linear (PL) function over the simplicial complex by linear interpolation; finally, to apply some exact computation algorithm for PL functions. This is the approach advocated by Dey and Wang [64], who rely on the $O(n \log n)$ expected time algorithm of Harvey, Wenger and Wang [81] for the last step. The drawbacks of this approach are:

- Its relative complexity: the Reeb graph computation from the PL function is based on collapses of its simplicial domain that may break the complex structure temporarily and therefore require some repairs.
- Its overall computational cost: here, n is not the number of data points, but the number of vertices, edges and triangles of the simplicial complex, which, in principle, can be up to cubic in the number of data points if we use a neighborhood graph. Indeed, the triangles are needed to compute an approximation of the Reeb graph, in the same way as they are to compute 1-dimensional homology.

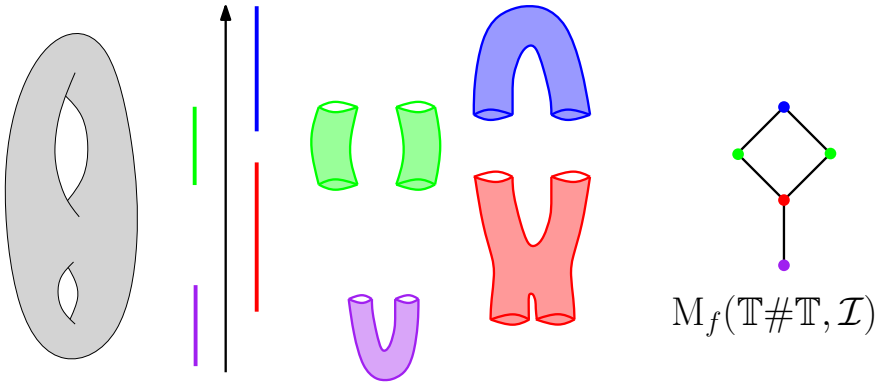

Figure 2.16: Example of the Mapper computed on the double torus $\mathbb{T}\#\mathbb{T}$ with the height function f. The cover $\mathcal{I}$ of $\mathrm{im}(f) \subseteq \mathbb{R}$ has four intervals (red, green, blue and purple), and the cover of the double torus has five connected components (one is blue, one is red, one is purple and the other two are green). The Mapper is displayed on the right.

2.5 Mapper

To cope with the computational issue of the Reeb graph, the *Mapper* was introduced by Singh, Mémoli and Carlsson [129] as a discrete version of the Reeb graph. The main difference is that it requires to compute the connected components of preimages of *intervals* instead of singletons. In the case of point clouds, finding such connected components amounts to apply clustering methods on the preimages. For this reason, and due to its success in many different applications [3, 5, 97, 108], the Mapper has become an emblematic tool of Topological Data Analysis.

It is defined in a formal way as the *nerve* of a specific *cover* of a topological space.

Covers and Nerves

Nerve of a cover. Let Z be a topological space. A *cover* of Z is a family $\mathcal{U}$ of subsets of Z, $\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$, such that $Z = \bigcup_{\alpha \in A} U_\alpha$. It is *open* if all its elements are open subspaces of Z. It is *connected* if all its elements are connected subspaces of Z. Its *nerve* is the abstract simplicial complex $\mathcal{N}(\mathcal{U})$ that has one k-simplex per $(k+1)$-fold intersection of elements of $\mathcal{U}$:

$$\{\alpha_0, ..., \alpha_k\} \in \mathcal{N}(\mathcal{U}) \iff \bigcap_{i=0,\dots,k} U_{\alpha_i} \neq \emptyset.$$

Generic and minimal cover. When a subfamily $\mathcal{V}$ of $\mathcal{U}$ is itself a cover of Z, it is called a *subcover* of $\mathcal{U}$. It is *proper* if it is not equal to $\mathcal{U}$. Finally, $\mathcal{U}$ is called *minimal* if it admits no proper subcover or, equivalently, if it has no element included in the union of the other elements. Given a minimal cover $\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$, for every $\alpha \in A$ we let

$$\tilde{U}_\alpha = U_\alpha \setminus \bigcup_{\alpha' \neq \alpha} U_{\alpha'} \cap U_\alpha,$$

be the *proper subset* of U_α, that is the maximal subset of U_α that has an empty intersection with the other elements of $\mathcal{U}$. $\mathcal{U}$ is called *generic* if no connected component of the proper subsets of its elements is a singleton.

Mapper

Let X, Z be topological spaces and let $f : X \to Z$ be a continuous function. Consider a cover $\mathcal{U}$ of $\mathrm{im}(f)$, and pull it back to X via f^{-1}. Then, decompose every $V_\alpha = f^{-1}(U_\alpha) \subseteq X$ into its connected components: $V_\alpha = \bigsqcup_{i \in \{1 \dots c(\alpha)\}} V_\alpha^i$, where $c(\alpha)$ is the number of connected components of V_α. Then, $\mathcal{V} = \{V_\alpha^i\}_{\alpha \in A, i \in \{1, \cdots, c(\alpha)\}}$ is a connected cover of X. It is called the *connected pullback cover*, and its nerve $\mathcal{N}(\mathcal{V})$ is the Mapper.

Definition 2.5.1. *Let X, Z be topological spaces, $f : X \to Z$ be a continuous function, $\mathcal{U}$ be a cover of* $\mathrm{im}(f)$ *and $\mathcal{V}$ be the associated connected pullback cover.*

Then, the Mapper *of X is* $\mathrm{M}_f(X, \mathcal{U}) = \mathcal{N}(\mathcal{V})$.

See Figure 2.16 for an illustration. Note that the Mapper is a simplicial complex and, as a combinatorial object, does not contain metric information. In particular, its edges have no associated lengths. We recall that when the space X is a point cloud, the connected pullback cover is computed with clustering. We study this discrete case in more depth in Chapter 5, where we use single-linkage clustering.

Computation

The construction of Mappers from point cloud data is very easy to describe and to implement, using standard clustering methods to detect connected components. For instance, if single-linkage clustering is used, it only requires to build the edges of a single neighborhood graph, whose size scales up at worst quadratically (and not cubically) with the size of the input point cloud.

CHAPTER 3

TELESCOPES AND REEB GRAPHS

In this chapter, we study connections between the metrics between Reeb graphs presented in Section 2.4.2. We recall that these metrics either enjoy good properties—like stability or discriminativity—but are intractable to compute, such as the functional distortion distance d_{FD}, or they are computable but lack discriminative power globally, such as the bottleneck distance d_{b} between the extended persistence diagrams of the Reeb graphs. The main result of this chapter is Theorem 3.0.1, which states that d_{b} is actually *locally* equivalent to d_{FD}, in some specific sense of locality.

Indeed, since the bottleneck distance is only a pseudometric—see Figure 2.14, the inequality given by Theorem 2.4.13 in Chapter 2 cannot be turned into a global equivalence result. However, for any pair of Reeb graphs $\mathrm{R}_f(X)$ and $\mathrm{R}_g(Y)$ that have the same extended persistence diagrams $\mathrm{ExDg}(\tilde{f}) = \mathrm{ExDg}(\tilde{g})$, and that are at positive functional distortion distance from each other $d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) > 0$, every continuous path in d_{FD} from $\mathrm{R}_f(X)$ to $\mathrm{R}_g(Y)$ will perturb the points of $\mathrm{ExDg}(\tilde{f})$ and eventually drive them back to their initial position, suggesting first that d_{b} may be locally equivalent to d_{FD}—which is the main result of this chapter, but also that, even though $d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) = 0$, the intrinsic metric $\hat{d}_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{R}_g(Y))$ induced by d_{b} may be positive—which we state in Theorem 3.3.2.

Local equivalence. Let X, Y be topological spaces and $f : X \to \mathbb{R}$, $g : Y \to \mathbb{R}$ be Morse-type functions. Let $\mathrm{Crit}(f) = \{a_1, \cdots, a_n\}$ and $\mathrm{Crit}(g) = \{b_1, \cdots, b_m\}$, $n, m \in \mathbb{N}^*$, be the critical values of f and g respectively. Finally, let $a_f = \min\{a_{i+1} - a_i : 1 \leq i \leq n-1\} > 0$ and $a_g = \min\{b_{j+1} - b_j : 1 \leq j \leq m-1\} > 0$ be the minimal distances between consecutive critical values of f or g. In this chapter, we will show the following local equivalence theorem:

Theorem 3.0.1. *Let $K \in (0, 1/22]$. If $d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq \max\{a_f, a_g\}/(8(1+22K))$, then:*

$$\begin{aligned} &K d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq K d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \\ &\leq d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \\ &\leq 2\, d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)) \leq 6\, d_{\mathrm{fGH}}(\mathrm{R}_f(X), \mathrm{R}_g(Y)). \end{aligned} \tag{3.1}$$

Note that the notion of locality used here is slightly different from the usual one. On the one hand, the equivalence does not hold for any arbitrary pair of Reeb graphs inside a neighborhood of some fixed Reeb graph, but rather for any pair involving the fixed graph. On the other hand, the constants in the equivalence are independent of the pair of Reeb graphs considered.

To prove this result, we use the so-called *telescope* structure of the Reeb graphs. The Reeb graph is known to be a graph (technically, a multi-graph) when X is a smooth manifold and f is a Morse function, or more generally when f is of Morse type, as in Definition 2.3.3. In that case, the Reeb graph can be decomposed into edges glued together at critical levels. This can be generalized into the so-called *telescopes*, which are adjunction topological spaces that can be decomposed into cylinders glued together at specific levels termed "critical". This telescope decomposition allows to define several operators acting on the critical levels that we use to prove the local equivalence.

Plan of the Chapter. We give the formal definition of telescopes in Section 3.1. We also use this decomposition to define several telescope operators, which will also be used later in this thesis, such as in Chapters 4 and 5. Next, using the telescope structure of Reeb graphs, we show Theorem 3.0.1 in Section 3.2. Finally, we end the chapter with Section 3.3, in which we study the intrinsic metrics that are *induced* by the metrics of Section 2.4, and show that they are all equivalent.

Convention. In this thesis, we work with singular homology with coefficients in $\mathbb{Z}_2$, which we omit in our notations for simplicity, and we use the term "connected" as a shorthand for "path-connected".

3.1 Telescopes and Operators

Recall that, given topological spaces X and $A \subseteq Y$ together with a continuous map $f : A \to X$, the *adjunction space* $X \cup_f Y$ (also denoted $Y \cup_f X$) is the quotient of the disjoint union $X \amalg Y$ by the equivalence relation induced by the identifications $\{f(a) \sim a\}_{a \in A}$.

Definition 3.1.1 (Telescope [28])**.** *A* telescope *is an adjunction space of the following form:*

$$T = (Y_0 \times (a_0, a_1]) \cup_{\psi_0} (X_1 \times \{a_1\}) \cup_{\phi_1} (Y_1 \times [a_1, a_2]) \cup_{\psi_1} ... \cup_{\phi_n} (Y_n \times [a_n, a_{n+1})),$$

where $-\infty = a_0 < a_1 < \cdots < a_n < a_{n+1} = +\infty$, and where the $\phi_i : Y_i \times \{a_i\} \to X_i \times \{a_i\}$ and $\psi_i : Y_i \times \{a_{i+1}\} \to X_{i+1} \times \{a_{i+1}\}$ are continuous maps. The a_i are called the critical values *of T and their set is denoted by* $\mathrm{Crit}(T)$*, the ϕ_i and ψ_i are called* attaching maps*, the Y_i are compact and locally connected spaces called the* cylinders *and the X_i are topological spaces called the* critical slices*. Moreover, all Y_i and X_i have finitely-generated homology.*

Extended persistence diagram. A telescope comes equipped with functions π_1 and π_2, which are the projections onto the first factor and second factor respectively. From now on, given any interval I, we let T^I denote $\pi_1 \circ \pi_2^{-1}(I)$. Then, the extended persistence diagram $\mathrm{ExDg}(\pi_2)$ can be described using the following Lemma.

Lemma 3.1.2. *Since ϕ_i and ψ_i are continuous,*

$$\forall \alpha \in [a_i, a_{i+1}),\ T^{(-\infty,\alpha]} \text{ deform retracts onto } T^{(-\infty,a_i]}$$
$$\forall \alpha \in (a_{i-1}, a_i],\ T^{[\alpha,+\infty)} \text{ deform retracts onto } T^{[a_i,+\infty)},$$

where a topological space X is said to deform retract *onto $Y \subseteq X$ if there exists a continuous function $F : X \times [0,1] \to X$ such that $F(\cdot, 0) = \mathrm{id}_X$, $F|_{Y \times \{\alpha\}}(\cdot, \alpha) = \mathrm{id}_Y$ for any $\alpha \in [0,1]$, and $F(X, 1) \subseteq Y$. In particular, this means that the inclusion $Y \hookrightarrow X$ is a homotopy equivalence.*

Corollary 3.1.3. *The following inclusion holds:* $\mathrm{ExDg}(\pi_2) \subseteq \mathrm{Crit}(T) \times \mathrm{Crit}(T)$.

Construction from a Morse-type function. One can build telescopes from the domain of Morse-type functions—see Definition 2.3.3. Indeed, a function $f : X \to \mathbb{R}$ of Morse type naturally induces a telescope $T(X, f)$ with

- $\mathrm{Crit}(T(X, f)) = \mathrm{Crit}(f)$,
- $X_i = f^{-1}(a_i)$,
- $Y_i = \pi_1 \circ \mu_i^{-1} \circ f^{-1}((a_i, a_{i+1}))$,
- $\phi_i : (y, a_i) \mapsto (\bar{\mu}_i|_{Y_i \times \{a_i\}}(y, a_i), a_i)$, $\forall y \in Y_i$, $\forall i \in \{1, ..., n\}$,
- $\psi_i : (y, a_{i+1}) \mapsto (\bar{\mu}_i|_{Y_i \times \{a_{i+1}\}}(y, a_{i+1}), a_{i+1})$, $\forall y \in Y_i$, $\forall i \in \{0, ..., n-1\}$,

$T(X, f)$ is well-defined thanks to the following Lemma:

Lemma 3.1.4. $\mathrm{im}(\phi_i) \subseteq f^{-1}(a_i) \times \{a_i\}$ *and* $\mathrm{im}(\psi_i) \subseteq f^{-1}(a_{i+1}) \times \{a_{i+1}\}$.

Proof. Let $(y, a_{i+1}) \in Y_i \times \{a_{i+1}\}$. Consider the sequence $(y, v_n)_{n\in\mathbb{N}}$, for an arbitrary $(v_n)_{n\in\mathbb{N}} \in (a_i, a_{i+1})^{\mathbb{N}}$ that converges to a_{i+1}. Then, $(f\circ\bar{\mu}_i(y, v_n))_{n\in\mathbb{N}}$ converges to $f\circ\bar{\mu}_i(y, a_{i+1})$ by continuity of $f\circ\bar{\mu}$. Moreover, for all $n \in \mathbb{N}$ we have $f \circ \bar{\mu}_i(y, v_n) = f \circ \mu_i(y, v_n) = v_n$ since $f|_{f^{-1}(a_i,a_{i+1})} = \pi_2 \circ \mu_i^{-1}$. Therefore, $(f \circ \bar{\mu}_i(y, v_n))_{n\in\mathbb{N}}$ converges also to a_{i+1}. By uniqueness of the limit, we have $f \circ \bar{\mu}_i(y, a_{i+1}) = a_{i+1}$, meaning that $\bar{\mu}_i(y, a_{i+1}) \in f^{-1}(a_{i+1})$. Thus, $\mathrm{im}(\psi_i) \subseteq f^{-1}(a_{i+1}) \times \{a_{i+1}\}$. The same argument applies to show that $\mathrm{im}(\phi_i) \subseteq f^{-1}(a_i) \times \{a_i\}$. □

Correspondence between X and $T(X, f)$. We now exhibit a homeomorphism between $T(X, f)$ and X. Let $\mu : T(X, f) \to X$ be defined by:

$$\mu(y, z) = \begin{cases} y \text{ if } (y, z) \in X_i \times \{a_i\} \text{ for some } i; \\ \mu_i(y, z) \text{ if } (y, z) \in Y_i \times (a_i, a_{i+1}) \text{ for some } i. \end{cases}$$

The map μ is bijective as every μ_i is. It is also continuous as every $\bar{\mu}_i$ is. Since every continuous bijection from a compact space to a Hausdorff space is a homeomorphism (see e.g. Proposition 13.26 in [131]), μ defines a homeomorphism between $T(X, f)$ and X. Moreover, $\pi_2 = f \circ \mu$ so $\mathrm{ExDg}(f) = \mathrm{ExDg}(\pi_2)$.

Operators on telescopes

The decomposition of telescopes into cylinders can be used to define simple operators that modify the telescope structures in a predictable way. Specifically, we detail three types of operators, corresponding to the cases where one asks for either removal of critical values (Merge operator), duplication of critical values (Split operator), or translation of critical values (Shift operator). To formalize this, we use *generalized attaching maps*:

$$\begin{array}{llll} \phi_i^a : & Y_i \times \{a\} \to & X_i \times \{a\}; & (y, a) \mapsto (\pi_1 \circ \phi_i(y, a_i), a), \\ \psi_i^a : & Y_i \times \{a\} \to & X_{i+1} \times \{a\}; & (y, a) \mapsto (\pi_1 \circ \psi_i(y, a_{i+1}), a). \end{array}$$

Merge

Merge operators merge all critival values of a telescope located in $[a, b]$ into a single critical value $\bar{a} = \frac{a+b}{2}$.

Definition 3.1.5 (Merge)**.** *Let T be a telescope. Let $a \leq b$. If $[a, b]$ contains at least one critical value, i.e. $\exists i, j \in \mathbb{N}$ such that $a_{i-1} < a \leq a_i \leq a_j \leq b < a_{j+1}$, then the* Merge *on T between a, b is the telescope $T' = \mathrm{Merge}_{a,b}(T)$ given by:*

$$\begin{gathered} ...(Y_{i-1} \times [a_{i-1}, a_i]) \cup_{\psi_{i-1}} (X_i \times \{a_i\}) \cup_{\phi_i} ... \cup_{\psi_{j-1}} (X_j \times \{a_j\}) \cup_{\phi_j} (Y_j \times [a_j, a_{j+1}])... \\ \downarrow \\ ...(Y_{i-1} \times [a_{i-1}, \bar{a}]) \cup_{f_{i-1}} (T^{[a,b]} \times \{\bar{a}\}) \cup_{g_j} (Y_j \times [\bar{a}, a_{j+1}])... \end{gathered}$$

where $\bar{a} = \frac{a+b}{2}$, where $f_{i-1} = \psi_{i-1}^{\bar{a}}$ if $a = a_i$ and $f_{i-1} = \mathrm{id}_{Y_{i-1}\times\{\bar{a}\}}$ otherwise, and where $g_j = \phi_j^{\bar{a}}$ if $b = a_j$ and $g_j = \mathrm{id}_{Y_j\times\{\bar{a}\}}$ otherwise.

If $[a, b]$ contains no critical value, i.e. $a_{i-1} < a \leq b < a_i$, then $\mathrm{Merge}_{a,b}(T)$ is given by:

$$\begin{gathered} ...(X_{i-1} \times \{a_{i-1}\}) \cup_{\phi_{i-1}} (Y_{i-1} \times [a_{i-1}, a_i]) \cup_{\psi_{i-1}} (X_i \times \{a_i\})... \\ \downarrow \\ ... \cup_{\phi_{i-1}} (Y_{i-1} \times [a_{i-1}, \bar{a}]) \cup_{f_{i-1}} (T^{[a,b]} \times \{\bar{a}\}) \cup_{g_{i-1}} (Y_{i-1} \times [\bar{a}, a_i]) \cup_{\psi_{i-1}} ... \end{gathered}$$

where $\bar{a} = \frac{a+b}{2}$, and where $f_{i-1} = g_{i-1} = \mathrm{id}_{Y_{i-1}\times\{\bar{a}\}}$.

See the left panel of Figure 3.1 for an illustration.

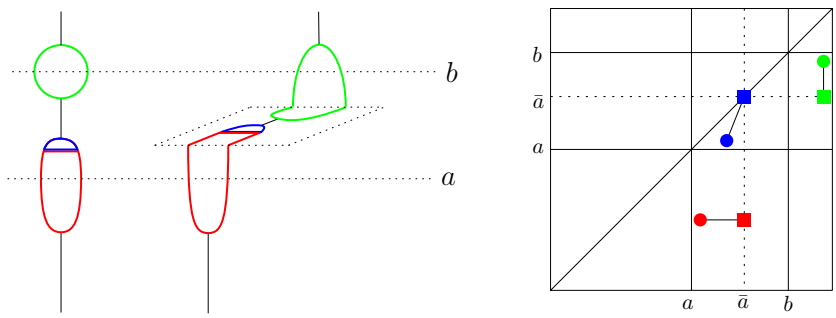

Figure 3.1: Left: Effect of a Merge on a telescope. Right: Effect on the corresponding extended persistence diagram. Points before the Merge are disks while points after the Merge are squares.

Merge for persistence diagrams. Similarly, we define the Merge between a, b on an extended persistence diagram ExDg as the diagram $\mathrm{Merge}_{a,b}(\mathrm{ExDg})$ given by $\mathrm{Merge}_{a,b}(x, y) = (\bar{x}, \bar{y})$, where:

$$\bar{x} = \begin{cases} x \text{ if } x \notin [a, b] \\ \bar{a} \text{ otherwise} \end{cases} \quad \text{and } \bar{y} = \begin{cases} y \text{ if } y \notin [a, b] \\ \bar{a} \text{ otherwise} \end{cases}$$

Points in the strips $x \in [a, b]$, $y \in [a, b]$ are snapped to the lines $x = \bar{a}$ and $y = \bar{a}$ respectively. See the right panel of Figure 3.1. See also the first intermediate points along the trajectories of the red points in Figure 4.9 for another illustration on extended persistence diagrams.

Commutativity of the operators. We now prove that extended persistent homology commutes with this operator, i.e. $\mathrm{ExDg}(\mathrm{Merge}) = \mathrm{Merge}(\mathrm{ExDg})$.

Lemma 3.1.6. *Let $a \leq b$ and $T' = \mathrm{Merge}_{a,b}(T)$. Let $\pi'_2 : T' \to \mathbb{R}$ be the projection onto the second factor. Then, $\mathrm{ExDg}(\pi'_2) = \mathrm{Merge}_{a,b}(\mathrm{ExDg}(\pi_2))$.*

Proof. We only study the sublevel sets of the functions, which means that we only prove the result for the ordinary part of the diagrams. The proof is symmetric for superlevel sets, leading to the result for the extended and the relative parts.

Assume $a_{i-1} < a \leq a_i \leq a_j \leq b < a_{j+1}$. Given $x \leq y$, we let $\Pi_{x,y} : H_*\left(T^{(-\infty,x]}\right) \to H_*\left(T^{(-\infty,y]}\right)$ and $\Pi'_{x,y} : H_*\left((T')^{(-\infty,x]}\right) \to H_*\left((T')^{(-\infty,y]}\right)$ be the homomorphisms induced by inclusions. Since f is of Morse type, Lemma 3.1.2 relates Π' to Π as follows (see Figure 3.2):

$$\Pi'_{x,y} = \begin{cases} \Pi_{x,y} \text{ if } x, y \notin [a, b] \text{ (green)} & \Pi_{x,a_{i-1}} \text{ if } x < a, y \in [a, \bar{a}) \text{ (pink)} \\ \Pi_{a_{i-1},y} \text{ if } x \in [a, \bar{a}), y > b \text{ (blue)} & \Pi_{a_{i-1},a_j} \text{ if } x \in [a, \bar{a}), y \in [\bar{a}, b] \text{ (orange)} \\ \Pi_{a_j,y} \text{ if } x \in [\bar{a}, b], y > b \text{ (grey)} & \mathrm{id}^*_{Y_{i-1}} \text{ if } x, y \in [a, \bar{a}) \text{ (brown)} \\ \Pi_{x,a_j} \text{ if } x < a, y \in [\bar{a}, b] \text{ (turquoise)} & \mathrm{id}^*_{Y_j} \text{ if } x, y \in [\bar{a}, b] \text{ (purple)} \end{cases} \tag{3.2}$$

The equality between the diagrams follows from these relations and the inclusion-exclusion formula (2.6). Consider for instance the case where the point $(x, y) \in \mathrm{ExDg}(\pi_2)$ belongs to the union A of the pink and the turquoise areas. One can select two abscissae $x_1 < x < x_2$ and an arbitrarily small $\epsilon > 0$. Then, the total multiplicity of the corresponding rectangle R in $\mathrm{ExDg}(\pi'_2)$ (displayed in the right panel of Figure 3.2) is given by:

$$\mathrm{mult}(R) = \mathrm{rank}\ \Pi'_{x_2,a-\epsilon} - \mathrm{rank}\ \Pi'_{x_2,b+\epsilon} + \mathrm{rank}\ \Pi'_{x_1,b+\epsilon} - \mathrm{rank}\ \Pi'_{x_1,a-\epsilon}.$$

The first relation in (3.2) shows that R has exactly the same multiplicity in $\mathrm{ExDg}(\pi_2)$, since all its corners belong to the green area. As this is true for arbitrarily small $\epsilon > 0$, it means that $R' = R \cap A$ also has the same multiplicity in $\mathrm{ExDg}(\pi_2)$ as in $\mathrm{ExDg}(\pi'_2)$. Now, if we pick a point inside R' with an ordinate different than $\bar{a}$, we can compute its multiplicity in $\mathrm{ExDg}(\pi'_2)$ by surrounding it with a box included in the turquoise area (if the ordinate is bigger than $\bar{a}$) or in the pink area (if it is smaller).

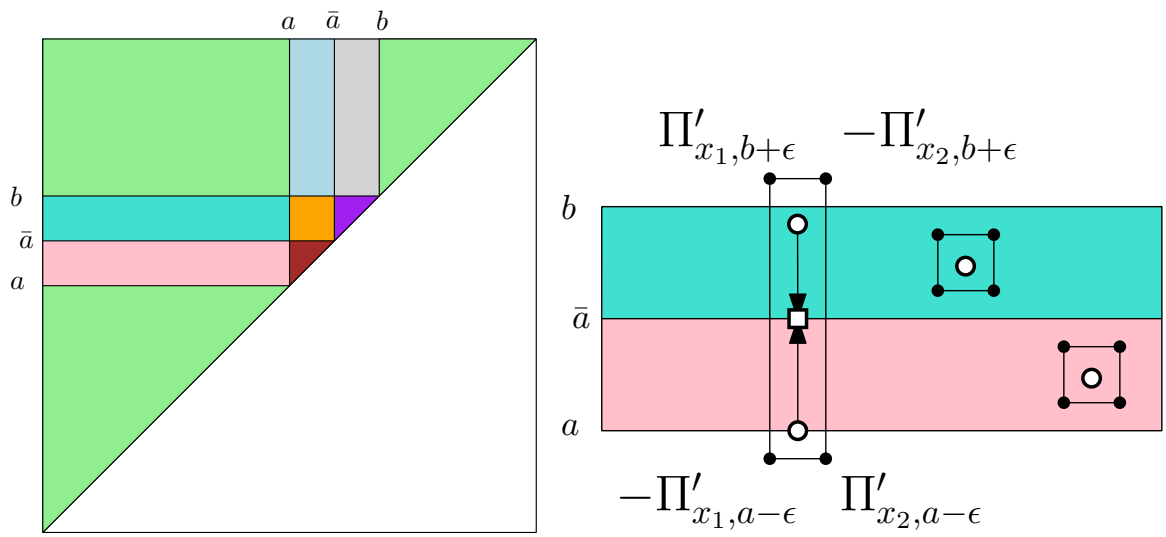

Figure 3.2: Left: Areas of the extended persistence diagram used in the proof. Right: Examples of the boxes we use to prove the result (circles represent points before the Merge, squares represent points after the Merge).

Boxes in the turquoise area have multiplicity $\operatorname{rank} \Pi'_{x_2,y_1} - \operatorname{rank} \Pi'_{x_2,y_2} + \operatorname{rank} \Pi'_{x_1,y_2} - \operatorname{rank} \Pi'_{x_1,y_1} = \operatorname{rank} \Pi_{x_2,a_j} - \operatorname{rank} \Pi_{x_2,a_j} + \operatorname{rank} \Pi_{x_1,a_j} - \operatorname{rank} \Pi_{x_1,a_j} = 0$. Similarly, boxes in the pink area also have multiplicity zero. Thus, all points of R' in $\mathrm{ExDg}(\pi'_2)$ have ordinate $\bar{a}$. Again, as it is true for x_1, x_2 as close to each other as we want, it means that (x, y) is snapped to $(x, \bar{a})$ in $\mathrm{ExDg}(\pi'_2)$. The treatment of the other areas in the plane is similar.

Now, if $[a, b]$ contains no critical values, then $\Pi' = \Pi$, so the result is clear. □

Split

Split operators split a critical value a_i into two different ones $a_i - \varepsilon$ and $a_i + \varepsilon$.

Definition 3.1.7 (Split)**.** *Let T be a telescope. Let $a_i \in \mathrm{Crit}(T)$ and ϵ such that*

$$0 \leq \epsilon < \min\{a_{i+1} - a_i, a_i - a_{i-1}\}.$$

The ϵ-Split on T at a_i is the telescope $T' = \mathrm{Split}_{\epsilon,a_i}(T)$ given by:

$$\begin{gathered}
...(Y_{i-1} \times [a_{i-1}, a_i]) \cup_{\psi_{i-1}} (X_i \times \{a_i\}) \cup_{\phi_i} (Y_i \times [a_i, a_{i+1}])... \\
\downarrow \\
...(Y_{i-1} \times [a_{i-1}, a_i - \epsilon]) \cup_{\psi_{i-1}^{a_i-\epsilon}} (X_i \times \{a_i - \epsilon\}) \cup_{\mathrm{id}} (X_i \times [a_i - \epsilon, a_i + \epsilon]) \cup_{\mathrm{id}} (X_i \times \{a_i + \epsilon\}) \cup_{\phi_i^{a_i+\epsilon}} (Y_i \times [a_i + \epsilon, a_{i+1}])...
\end{gathered}$$

See the left panel of Figure 3.3 for an illustration.

Down- and up-forks. Splits create particular critical values called *down-* and *up-forks*. Intuitively, Split operations allow to distinguish between all possible types of changes in 0- and 1-dimensional homology of the sublevel and superlevel sets, namely: union of two connected components, creation of a connected component, destruction of a connected component, and separation of a connected component. Unions and creations occur at down-forks while separations and destructions occur at up-forks. See Figure 3.4 for an illustration. We formalize and prove this intuition in Lemma 3.1.11.

Definition 3.1.8. *A critical value $a_i \in \mathrm{Crit}(T)$ is called an* up-fork *if ψ_{i-1} is an homeomorphism, and it is called a* down-fork *if ϕ_i is a homeomorphism.*

Since the attaching maps introduced by the Split are identity maps, we have the following lemma:

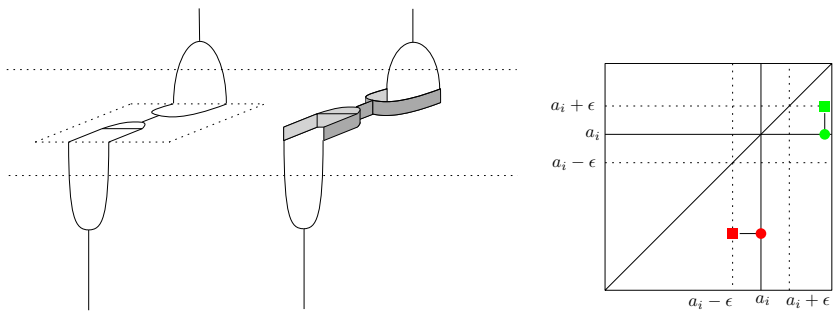

Figure 3.3: Left: Effect of a Split on a telescope. Right: Effect on the corresponding extended persistence diagram. Points before the Split are disks while points after the Split are squares.

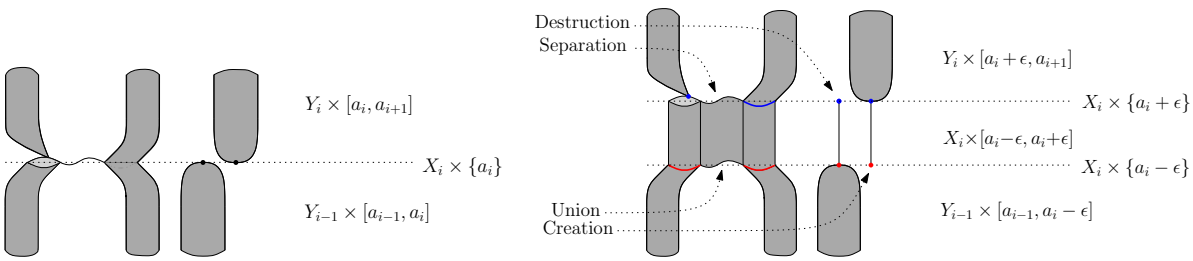

Figure 3.4: Left and right panels display the space before and after a Split respectively. Subsets of X_i that are colored in red and blue correspond to $\mathrm{im}(\pi_1 \circ \psi_{i-1})$ and $\mathrm{im}(\pi_1 \circ \phi_i)$ respectively.

Lemma 3.1.9. *The critical values $a_i-\epsilon$ and $a_i+\epsilon$ created with* Split *are down- and up-forks respectively.*

The next lemma is a direct consequence of the existence and continuity of ϕ_i^{-1} (resp. ψ_{i-1}^{-1}) when $a_i \in \mathrm{Crit}(T)$ is a down-fork (resp. up-fork):

Lemma 3.1.10. *Let $a_i \in \mathrm{Crit}(T)$. If a_i is an up-fork, then $T^{(-\infty,a_i]}$ deform retracts onto $T^{(-\infty,\alpha]}$ for all $\alpha \in (a_{i-1}, a_i]$. If a_i is a down-fork, then $T^{[a_i,+\infty)}$ deform retracts onto $T^{[\alpha,+\infty)}$ for all $\alpha \in [a_i, a_{i+1})$.*

Now we can prove the previous intuition concerning down- and up-forks correct:

Lemma 3.1.11. *Let $a_i \in \mathrm{Crit}(T)$. If a_i is an up-fork, then it can only be the birth time of relative cycles and the death time of relative and extended cycles in* $\mathrm{ExDg}(\pi_2)$*. If a_i is a down-fork, then it can only be the birth time of ordinary and extended cycles and the death time of ordinary cycles in* $\mathrm{ExDg}(\pi_2)$*.*

Proof. Let $0 \leq \epsilon, \epsilon' < \min\{a_{i+1}-a_i, a_i-a_{i-1}\}$. Consider the extended persistence module of π_2:

$$\begin{array}{ccccccccc} \ldots & \longrightarrow & H_*\left(T^{(-\infty,a_i-\epsilon]}\right) & \longrightarrow & H_*\left(T^{(-\infty,a_i]}\right) & \longrightarrow & H_*\left(T^{(-\infty,a_i+\epsilon']}\right) & \longrightarrow & \ldots \\ \ldots & \longrightarrow & H_*\left(T, T^{[a_i+\epsilon',+\infty)}\right) & \longrightarrow & H_*\left(T, T^{[a_i,+\infty)}\right) & \longrightarrow & H_*\left(T, T^{[a_i-\epsilon,+\infty)}\right) & \longrightarrow & \ldots \end{array}$$

If a_i is an up-fork, then the composition $H_*(T^{(-\infty,a_i-\epsilon]}) \to H_*(T^{(-\infty,a_i+\epsilon']})$ is an isomorphism since $T^{(-\infty,a_i+\epsilon']}$ deform retracts onto $T^{(-\infty,a_i-\epsilon]}$ by Lemmas 3.1.2 and 3.1.10. As ϵ, ϵ' can be chosen arbitrarily small, there cannot be any creation of ordinary or extended cycle at a_i. There also cannot be any destruction of ordinary cycle.

Similarly, if a_i is a down-fork, then the composition $H_*(T, T^{[a_i+\epsilon',+\infty)}) \to H_*(T, T^{[a_i-\epsilon,+\infty)})$ is an isomorphism since $T^{[a_i-\epsilon,+\infty)}$ deform retracts onto $T^{[a_i+\epsilon',+\infty)}$. Again, there cannot be any destruction of extended or relative cycle at a_i. There also cannot be any creation of relative cycle. $\square$

Split for persistence diagrams. Similarly, we define the ϵ-Split at a_i on a diagram ExDg as the diagram $\mathrm{Split}_{\epsilon,a_i}(\mathrm{ExDg})$ given by $\mathrm{Split}_{\epsilon,a_i}(x,y) = (\bar{x},\bar{y})$, where:

$$\bar{x} = \begin{cases} x \text{ if } x \neq a_i \\ a_i + \epsilon \text{ if } x = a_i \text{ and } (x,y) \in \mathrm{Rel} \\ a_i - \epsilon \text{ if } x = a_i \text{ and } (x,y) \notin \mathrm{Rel} \end{cases} \quad \text{and } \bar{y} = \begin{cases} y \text{ if } y \neq a_i \\ a_i - \epsilon \text{ if } y = a_i \text{ and } (x,y) \in \mathrm{Ord} \\ a_i + \epsilon \text{ if } y = a_i \text{ and } (x,y) \notin \mathrm{Ord} \end{cases}$$

Points located on the lines $x, y = a_i$ are snapped to the lines $x, y = a_i \pm \epsilon$ according to their type. Note that the definition of $\mathrm{Split}_{\epsilon,a_i}(\mathrm{ExDg})$ assumes implicitly that ExDg contains no point within the horizontal and vertical bands $[a_i - \epsilon, a_i) \times \mathbb{R}$, $(a_i, a_i + \epsilon] \times \mathbb{R}$, $\mathbb{R} \times [a_i - \epsilon, a_i)$ and $\mathbb{R} \times (a_i, a_i + \epsilon]$, which is the case under the assumptions of Definition 3.1.7. See the right panel of Figure 3.3 for an illustration. See also the second intermediate points along the trajectories of the red points in Figure 4.9 for another illustration on extended persistence diagrams.

Commutativity of the operators. We now prove that extended persistent homology commutes with this operator, i.e. $\mathrm{ExDg}(\mathrm{Split}) = \mathrm{Split}(\mathrm{ExDg})$.

Lemma 3.1.12. *Let $a_i \in \mathrm{Crit}(T)$. Let $0 < \epsilon < \min\{a_{i+1} - a_i, a_i - a_{i-1}\}$, $T' = \mathrm{Split}_{\epsilon,a_i}(T)$ and $\pi_2' : T' \to \mathbb{R}$ the projection onto the second factor. Then, $\mathrm{ExDg}(\pi_2') = \mathrm{Split}_{\epsilon,a_i}(\mathrm{ExDg}(\pi_2))$.*

Proof. Note that $T = \mathrm{Merge}_{a_i-\epsilon,a_i+\epsilon}(T')$. Hence, by Lemma 3.1.6, $\mathrm{ExDg}(\pi_2)$ can be obtained from $\mathrm{ExDg}(\pi_2')$ with $\mathrm{ExDg}(\pi_2) = \mathrm{Merge}_{a_i-\epsilon,a_i+\epsilon}(\mathrm{ExDg}(\pi_2'))$. Note also that π_2' has no critical value within the open interval $(a_i - \epsilon, a_i + \epsilon)$, so $\mathrm{ExDg}(\pi_2')$ has no point within the horizontal and vertical bands $\mathbb{R} \times (a_i - \varepsilon, a_i + \varepsilon)$ and $(a_i - \varepsilon, a_i + \varepsilon) \times \mathbb{R}$. Finally, Lemma 3.1.9 ensures that $a_i + \epsilon, a_i - \epsilon$ are up- and down-forks respectively, so Lemma 3.1.11 tells us exactly where the preimages of the points of $\mathrm{ExDg}(\pi_2)$ through the Merge are located depending on their type. □

Shift

Shift operators translate critical values.

Definition 3.1.13 (Shift)**.** *Let T be a telescope. Let $a_i \in \mathrm{Crit}(T)$ and ϵ such that*

$$0 \leq |\epsilon| < \min\{a_{i+1} - a_i, a_i - a_{i-1}\}.$$

The ϵ-Shift on T at a_i is the telescope $T' = \mathrm{Shift}_{\epsilon,a_i}(T)$ given by:

$$\begin{array}{c} ...(Y_{i-1} \times [a_{i-1}, a_i]) \cup_{\psi_{i-1}} (X_i \times \{a_i\}) \cup_{\phi_i} (Y_i \times [a_i, a_{i+1}])... \\ \downarrow \\ ...(Y_{i-1} \times [a_{i-1}, a_i + \epsilon]) \cup_{\psi_{i-1}^{a_i+\epsilon}} (X_i \times \{a_i + \epsilon\}) \cup_{\phi_i^{a_i+\epsilon}} (Y_i \times [a_i + \epsilon, a_{i+1}])... \end{array}$$

See the left panel of Figure 3.5 for an illustration.

Shift for persistence diagrams. Similarly, we define the ϵ-Shift at a_i on a diagram ExDg as the diagram $\mathrm{Shift}_{\epsilon,a_i}(\mathrm{ExDg})$ given by $\mathrm{Shift}_{\epsilon,a_i}(x,y) = (\bar{x},\bar{y})$ where:

$$\bar{x} = \begin{cases} x \text{ if } x \neq a_i \\ a_i + \epsilon \text{ otherwise} \end{cases} \quad \text{and } \bar{y} = \begin{cases} y \text{ if } y \neq a_i \\ a_i + \epsilon \text{ otherwise} \end{cases}$$

Points located on the lines $x, y = a_i$ are snapped to the lines $x, y = a_i + \epsilon$. Note that the definition of $\mathrm{Shift}_{\epsilon,a_i}(\mathrm{ExDg})$ assumes implicitly that ExDg contains no point within the horizontal and vertical bands delimited by a_i and $a_i + \varepsilon$, which is the case under the assumptions of Definition 3.1.13. See the right panel of Figure 3.5 for an illustration. See also the third intermediate points along the trajectories of the red points in Figure 4.9 for another illustration on extended persistence diagrams.

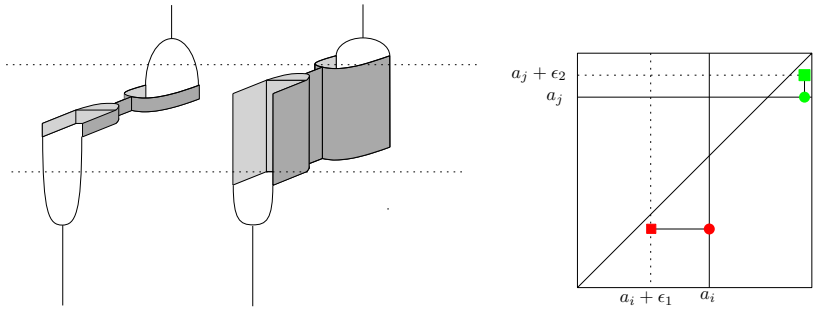

Figure 3.5: Left: Effect of a double Shift with amplitudes $\epsilon_1 < 0 < \epsilon_2$. Right: Effect on the corresponding extended persistence diagram. Points before the Shift are disks while points after the Shift are squares.

Commutativity of the operators. We now prove that extended persistent homology commutes with this operator, i.e. $\mathrm{ExDg}(\mathrm{Shift}) = \mathrm{Shift}(\mathrm{ExDg})$.

Lemma 3.1.14. *Let $a_i \in \mathrm{Crit}(T)$, $\epsilon \in (a_{i-1} - a_i,\ a_{i+1} - a_i)$, $T' = \mathrm{Shift}_{\epsilon,a_i}(T)$ and $\pi_2' : T' \to \mathbb{R}$ the projection onto the second factor. Then, $\mathrm{ExDg}(\pi_2') = \mathrm{Shift}_{\epsilon,a_i}(\mathrm{ExDg}(\pi_2))$.*

Proof. Again, the following relations coming from Lemma 3.1.2:

$$\Pi'_{x,y} = \begin{cases} \Pi_{x,y} \text{ if } x,y \notin (a_{i-1},a_{i+1}) \text{ (green)} & \Pi_{a_i,y} \text{ if } x \in [a_i+\epsilon, a_{i+1}), y \geq a_{i+1} \text{ (grey)} \\ \Pi_{x,a_{i-1}} \text{ if } x \leq a_{i-1}, y \in (a_{i-1}, a_i+\epsilon) \text{ (pink)} & \Pi_{a_{i-1},y} \text{ if } x \in (a_{i-1}, a_i+\epsilon), y \geq a_{i+1} \text{ (blue)} \\ \Pi_{x,a_i} \text{ if } x \leq a_{i-1}, y \in [a_i+\epsilon, a_{i+1}) \text{ (turquoise)} & \mathrm{id}^*_{Y_{i-1}} \text{ if } x,y \in (a_{i-1}, a_i+\epsilon) \text{ (brown)} \\ \Pi_{a_{i-1},a_i} \text{ if } x \in (a_{i-1}, a_i+\epsilon), y \in [a_i+\epsilon, a_{i+1}) \text{ (orange)} & \mathrm{id}^*_{Y_i} \text{ if } x,y \in [a_i+\epsilon, a_{i+1}) \text{ (purple)} \end{cases}$$

allow us to prove the result similarly to Lemma 3.1.6—see Figure 3.6. For instance, one can choose a box that intersects the lines $y = a_i + \epsilon$ and $y = a_i$, show that the total multiplicity is preserved, then choose another small box that does not intersect $y = a_i + \epsilon$ inside the first box, and show that its multiplicity is zero.

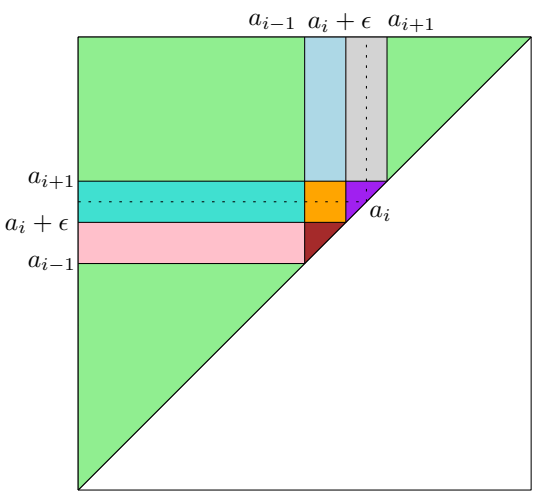

Figure 3.6: Areas of the extended persistence diagram used in the proof, with $\varepsilon < 0$.

□

3.2 A lower bound on d_{b}

In this section, we build on the Merge operator defined in the previous section and on the simplification operator defined in Section 2.4.3 to prove Theorem 3.0.1. Note that the upper bound in this theorem is

given by Theorem 2.4.13 and always holds. The aim of this section is to prove the lower bound.

Notation. Henceforth, we write R_f and R_g instead of $\mathrm{R}_f(X)$ and $\mathrm{R}_g(Y)$ to avoid heavy notations. We also assume without loss of generality that $\max\{a_f, a_g\} = a_f$ and we let $\varepsilon = d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_g)$.

Proof of Theorem 3.0.1 Let $K \in (0, 1/22]$. The proof proceeds by contradiction. Assuming that $d_{\mathrm{b}}(\mathrm{R}_f, \mathrm{R}_g) < K\varepsilon$, where $\varepsilon = d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_g) < a_f/(8(1+22K))$, it progressively transforms R_g into some other Reeb graph $\mathrm{R}_{g'}$ (Definition 3.2.1) that satisfies both $d_{\mathrm{FD}}(\mathrm{R}_g, \mathrm{R}_{g'}) < 22K\varepsilon \leq \varepsilon$ (Proposition 3.2.3) and $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) = 0$ (Proposition 3.2.4). The contradiction follows then from the triangle inequality.

Graph Transformation

The graph transformation is defined as the composition of the *Merge operator* from Section 3.1 and the *simplification operator* from Section 2.4.3.

Definition 3.2.1. *Let R_f be a fixed Reeb graph with critical values $\{a_1, \cdots, a_n\}$. Given $\alpha > 0$, the full transformation F_α is defined as*

$$F_\alpha = \mathrm{Merge}_{9\alpha} \circ S_{2\alpha},$$

where $\mathrm{Merge}_{9\alpha} = \mathrm{Merge}_{a_n - 9\alpha,\, a_n + 9\alpha} \circ \cdots \circ \mathrm{Merge}_{a_1 - 9\alpha,\, a_1 + 9\alpha}$.

See Figure 3.7 for an illustration of this smoothing transformation.

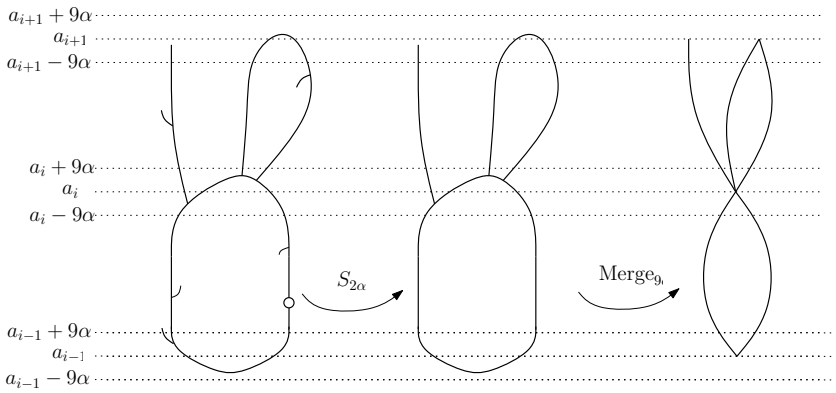

Figure 3.7: Illustration of F_α applied on an arbitrary Reeb graph.

Properties of the transformed graph

Let $\mathrm{R}_f, \mathrm{R}_g$ such that $d_{\mathrm{b}}(\mathrm{R}_f, \mathrm{R}_g) < K\varepsilon$ where $\varepsilon = d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_g) < a_f/(8(1+22K))$. Letting $\mathrm{R}_{g'} = F_{K\varepsilon}(\mathrm{R}_g)$, we want to show both that $d_{\mathrm{FD}}(\mathrm{R}_g, \mathrm{R}_{g'}) < 22K\varepsilon \leq \varepsilon$ (Proposition 3.2.3) and $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) = 0$ (Proposition 3.2.4), which will lead to a contradiction as mentioned previously. Let $B_\infty(x, r)$ denote the open ball of center x and radius $r > 0$ in the ℓ_∞-distance.

Lemma 3.2.2. *Let $\mathrm{R}_h = S_{2K\varepsilon}(\mathrm{R}_g)$. Under the above assumptions, one has*

$$\mathrm{ExDg}(h) \subset \bigcup\nolimits_{\tau \in \mathrm{ExDg}(f)} B_\infty(\tau, 9K\varepsilon). \tag{3.3}$$

Proof. Let $\mathrm{off}_{K\varepsilon}(\Delta) = \{x \in \mathbb{R}^2 : d_\infty(x, \Delta) \leq K\varepsilon\}$ be the $(K\varepsilon)$-offset of the diagonal Δ in the ℓ_∞-distance. Since $d_{\mathrm{b}}(\mathrm{R}_f, \mathrm{R}_g) < K\varepsilon$, we have $\mathrm{ExDg}(g) \subset \bigcup_{\tau \in \mathrm{ExDg}(f)} B_\infty(\tau, K\varepsilon) \cup \mathrm{off}_{K\varepsilon}(\Delta)$. Since $\mathrm{R}_h = S_{2K\varepsilon}(\mathrm{R}_g)$, it follows from Lemma 2.4.16 that $d_{\mathrm{b}}(\mathrm{ExDg}(h), \mathrm{ExDg}(g)) \leq 8K\varepsilon$. Moreover, since every persistence pair in $\mathrm{ExDg}(g) \cap \mathrm{off}_{K\varepsilon}(\Delta)$ is removed by $S_{2K\varepsilon}$, it results that:

$$\mathrm{ExDg}(h) \subset \bigcup\nolimits_{\tau \in \mathrm{ExDg}(g) \setminus \mathrm{off}_{K\varepsilon}(\Delta)} B_\infty(\tau, 8K\varepsilon) \subset \bigcup\nolimits_{\tau \in \mathrm{ExDg}(f)} B_\infty(\tau, 9K\varepsilon).$$

□

Now we can bound $d_{\mathrm{FD}}(\mathrm{R}_g, \mathrm{R}_{g'})$. Recall that, given an arbitrary Reeb graph R_h, with critical values $\mathrm{Crit}(h) = \{c_1, ..., c_p\}$, if C is a connected component of $h^{-1}(I)$, where I is an open interval such that $I \subseteq (c_i, c_{i+1})$ for some i, then C must be a *topological arc*, i.e. homeomorphic to an open interval.

Proposition 3.2.3. *Under the same assumptions as above, one has* $d_{\mathrm{FD}}(\mathrm{R}_g, \mathrm{R}_{g'}) < 22K\epsilon$.

Proof. Let $\mathrm{R}_h = S_{2K\varepsilon}(\mathrm{R}_g)$. The triangle inequality asserts that

$$d_{\mathrm{FD}}(\mathrm{R}_{g'}, \mathrm{R}_g) \leq d_{\mathrm{FD}}(\mathrm{R}_{g'}, \mathrm{R}_h) + d_{\mathrm{FD}}(\mathrm{R}_h, \mathrm{R}_g).$$

It suffices therefore to bound both $d_{\mathrm{FD}}(\mathrm{R}_{g'}, \mathrm{R}_h)$ and $d_{\mathrm{FD}}(\mathrm{R}_h, \mathrm{R}_g)$. By Lemma 2.4.16, we have $d_{\mathrm{FD}}(\mathrm{R}_h, \mathrm{R}_g) < 4K\varepsilon$. Now, recall from (3.3) that the points of the extended persistence diagram of R_h are included in $\bigcup_{\tau \in \mathrm{ExDg}(f)} B_\infty(\tau, 9K\varepsilon)$. Moreover, since $\mathrm{R}_{g'} = \mathrm{Merge}_{9K\varepsilon}(\mathrm{R}_h)$, $\mathrm{R}_{g'}$ and R_h are composed of the same number of arcs in each $[a_i + 9K\varepsilon, a_{i+1} - 9K\varepsilon]$. Hence, we can define explicit continuous maps $\phi : \mathrm{R}_h \to \mathrm{R}_{g'}$ and $\psi : \mathrm{R}_{g'} \to \mathrm{R}_h$ as depicted in Figure 3.8. More precisely, since R_h and $\mathrm{R}_{g'}$ are composed of the same number of arcs in each $[a_i + 9K\varepsilon, a_{i+1} - 9K\varepsilon]$, we only need to specify ϕ and ψ inside each interval $(a_i - 9K\varepsilon, a_i + 9K\varepsilon)$. Since the critical values of R_h are within distance less that $9K\varepsilon$ of the critical values of f, there exist two levels $a_i - 9K\varepsilon < \alpha_i \leq \beta_i < a_i + 9K\varepsilon$ such that R_h is only composed of arcs in $(a_i - 9K\varepsilon, \alpha_i]$ and $[\beta_i, a_i + 9K\varepsilon)$ for each i (dashed lines in Figure 3.8). For any connected component C of $h^{-1}((a_i - 9K\varepsilon, a_i + 9K\varepsilon))$, the map ϕ sends all points of $C \cap h^{-1}([\alpha_i, \beta_i])$ to the corresponding critical point y_C created by the Merge in $\mathrm{R}_{g'}$, and it maps the arcs of $C \cap h^{-1}((a_i - 9K\varepsilon, \alpha_i])$ and $C \cap h^{-1}([\beta_i, a_i + 9K\varepsilon))$ to the corresponding arcs in $\mathrm{R}_{g'}$. In return, the map ψ sends the critical point y_C to an arbitrary point of C. Then, since the Merge operation preserves connected components, for each arc A' of $(g')^{-1}((a_i - 9K\varepsilon, a_i + 9K\varepsilon))$ connected to y_C, there is at least one corresponding path A in R_h whose endpoint in $h^{-1}(a_i - 9K\varepsilon)$ or $h^{-1}(a_i + 9K\varepsilon)$ matches with the one of A' (see the colors in the second row of Figure 3.8). Hence ψ sends A' to A.
Let us bound the three terms in the $\max\{\cdots\}$ in (2.12) with this choice of maps ϕ, ψ:

- We first bound $\|h - g' \circ \phi\|_\infty$. Let $x \in \mathrm{R}_h$. Either $h(x) \in \bigcup_{i \in \{1,...,n-1\}} [a_i + 9K\varepsilon, a_{i+1} - 9K\varepsilon]$, and in this case we have $h(x) = g'(\phi(x))$ by definition of ϕ; or, there is $i_0 \in \{1, ..., n\}$ such that $h(x) \in (a_{i_0} - 9K\varepsilon, a_{i_0} + 9K\varepsilon)$ and then $g'(\phi(x)) \in (a_{i_0} - 9K\varepsilon, a_{i_0} + 9K\varepsilon)$. In both cases $|h(x) - g' \circ \phi(x)| < 18K\varepsilon$. Hence, $\|h - g' \circ \phi\|_\infty < 18K\varepsilon$.
- Since the previous proof is symmetric in h and g', one also has $\|g' - h \circ \psi\|_\infty < 18K\varepsilon$.
- We now bound $D(\phi, \psi)$. Let $(x, \phi(x)), (\psi(y), y) \in C(\phi, \psi)$ (the cases $(x, \phi(x)), (x', \phi(x'))$ and $(\psi(y), y), (\psi(y'), y')$ are similar). Let $\pi_{g'} : [0, 1] \to \mathrm{R}_{g'}$ be a continuous path from $\phi(x)$ to y which achieves $d_{g'}(\phi(x), y)$.
  - Assume $h(x) \in \bigcup_{i \in \{1,...,n-1\}} [a_i + 9K\varepsilon, a_{i+1} - 9K\varepsilon]$. Then one has $\psi \circ \phi(x) = x$. Hence, $\pi_h = \psi \circ \pi_{g'}$ is a valid path from x to $\psi(y)$. Moreover, since $\|g' - h \circ \psi\|_\infty < 18K\varepsilon$, it follows that
  $$\begin{aligned} \max\ \mathrm{im}(h \circ \pi_h) &< \max\ \mathrm{im}(g' \circ \pi_{g'}) + 18K\varepsilon, \\ \min\ \mathrm{im}(h \circ \pi_h) &> \min\ \mathrm{im}(g' \circ \pi_{g'}) - 18K\varepsilon. \end{aligned} \tag{3.4}$$
  Hence, one has
  $$\begin{aligned} d_h(x, \psi(y)) &\leq \max\ \mathrm{im}(h \circ \pi_h) - \min\ \mathrm{im}(h \circ \pi_h) < d_{g'}(\phi(x), y) + 36K\varepsilon, \\ -d_h(x, \psi(y)) &\geq \min\ \mathrm{im}(h \circ \pi_h) - \max\ \mathrm{im}(h \circ \pi_h) > -d_{g'}(\phi(x), y) - 36K\varepsilon. \end{aligned}$$
  This shows that $|d_h(x, \psi(y)) - d_{g'}(\phi(x), y)| < 36K\varepsilon$.
  - Assume that there is $i_0 \in \{1, ..., n\}$ such that $h(x) \in (a_{i_0} - 9K\varepsilon, a_{i_0} + 9K\varepsilon)$. Then, by definition of ϕ, ψ, we have $g'(\phi(x)) \in (a_{i_0} - 9K\varepsilon, a_{i_0} + 9K\varepsilon)$, and there is a path $\pi'_h : [0, 1] \to \mathrm{R}_h$ from x to $\psi \circ \phi(x)$ within the interval $(a_{i_0} - 9K\varepsilon, a_{i_0} + 9K\varepsilon)$, which itself is included in the interior of the offset $\mathrm{off}_{18K\varepsilon}(\mathrm{im}(g' \circ \pi_{g'}))$. Let now π_h be the concatenation of π'_h with $\psi \circ \pi_{g'}$, which goes from x to $\psi(y)$. Since $\|g' - h \circ \psi\| < 18K\varepsilon$, it follows that $\mathrm{im}(h \circ \psi \circ \pi_{g'}) \subseteq \mathrm{int}\ \mathrm{off}_{18K\varepsilon}(\mathrm{im}(g' \circ \pi_{g'}))$, and since $\mathrm{im}(h \circ \pi_h) = \mathrm{im}(h \circ \pi'_h) \cup \mathrm{im}(h \circ \psi \circ \pi_{g'})$ by concatenation, one finally has
  $$\mathrm{im}(h \circ \pi_h) \subseteq \mathrm{int}\ \mathrm{off}_{18K\varepsilon}(\mathrm{im}(g' \circ \pi_{g'})).$$
  Hence, the inequalities of (3.4) hold, implying that $|d_h(x, \psi(y)) - d_{g'}(\phi(x), y)| < 36K\varepsilon$.

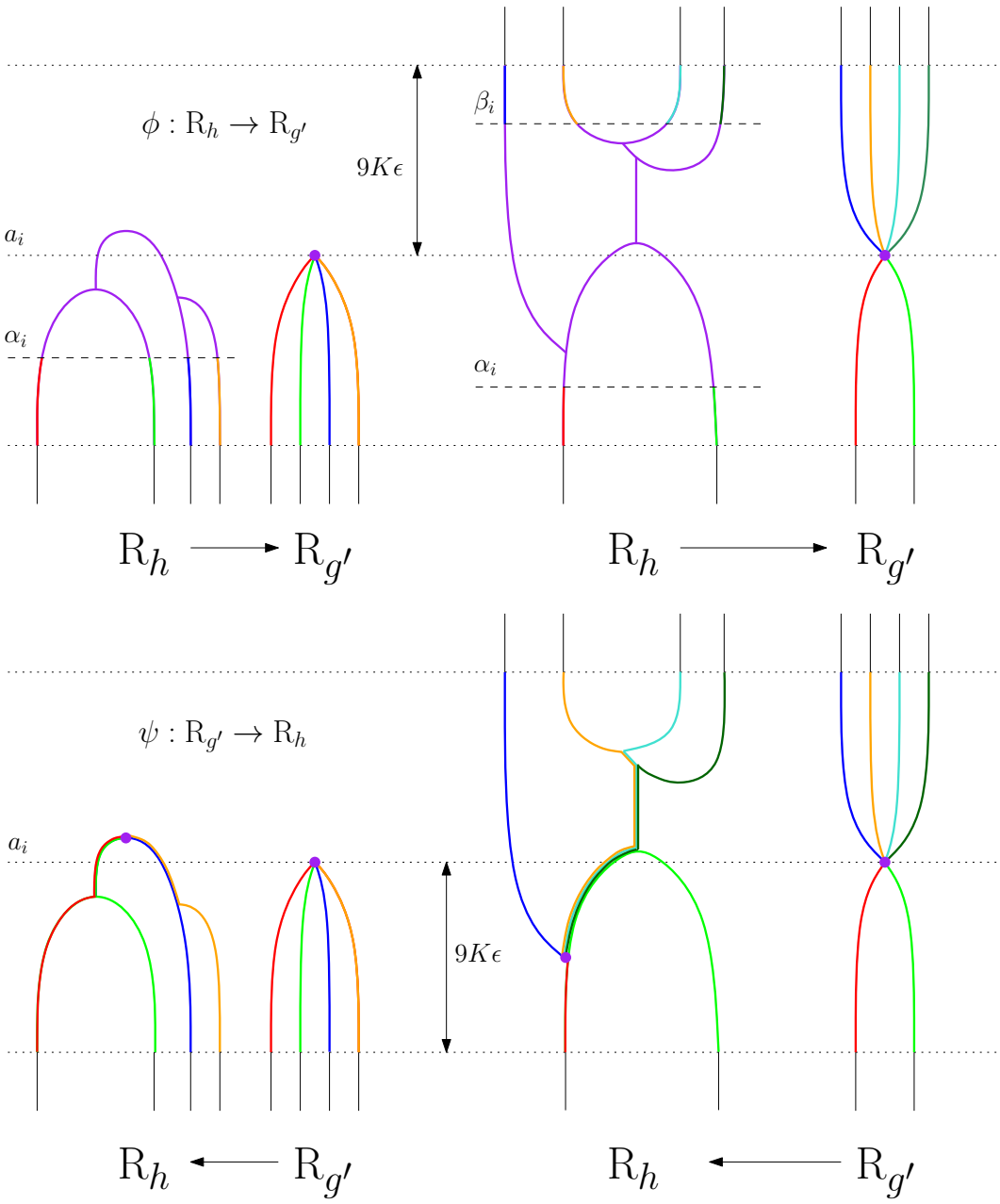

Figure 3.8: The effects of ϕ and ψ around a specific critical value a_i of f. Segments are matched according to their colors (up to reparameterization).

Since these inequalities hold for any $(x, \phi(x))$ and $(\psi(y), y)$, we deduce that $D(\phi, \psi) \leq 36K\varepsilon$. Thus, $d_{\mathrm{FD}}(\mathrm{R}_h, \mathrm{R}_g) < 4K\varepsilon$ and $d_{\mathrm{FD}}(\mathrm{R}_h, \mathrm{R}_{g'}) \leq 18K\varepsilon$, so $d_{\mathrm{FD}}(\mathrm{R}_{g'}, \mathrm{R}_g) < 22K\varepsilon$ as desired. □

To complete the proof, we now show that $\mathrm{R}_{g'}$ is isomorphic to R_f.

Proposition 3.2.4. *Under the same assumptions as above, one has* $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) = 0$.

Proof. First, recall from (3.3) that the points of the extended persistence diagram of R_h are included in $\bigcup_{\tau \in \mathrm{ExDg}(f)} B_\infty(\tau, 9K\varepsilon)$. Since $\mathrm{R}_{g'} = \mathrm{Merge}_{9K\varepsilon}(\mathrm{R}_h)$, it follows from Lemma 3.1.6 that $\mathrm{Crit}(g') \subseteq \mathrm{Crit}(f)$. Hence, both $\mathrm{R}_{g'}$ and R_f are composed of arcs in each (a_i, a_{i+1}).

Now, we show that, for each i, the number of arcs of $(g')^{-1}((a_i, a_{i+1}))$ and $f^{-1}((a_i, a_{i+1}))$ are the same. By the triangle inequality and Proposition 3.2.3, we have:

$$d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) \leq d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_g) + d_{\mathrm{FD}}(\mathrm{R}_g, \mathrm{R}_{g'}) < (1 + 22K)\varepsilon. \tag{3.5}$$

Let $\phi : \mathrm{R}_f \to \mathrm{R}_{g'}$ and $\psi : \mathrm{R}_{g'} \to \mathrm{R}_f$ be optimal continuous maps that achieve $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'})$. Let $i \in \{1, ..., n-1\}$. Assume that there are more arcs of $f^{-1}((a_i, a_{i+1}))$ than arcs of $(g')^{-1}((a_i, a_{i+1}))$. For every arc A of $f^{-1}((a_i, a_{i+1}))$, let $x_A \in A$ such that $f(x_A) = \bar{a} = \frac{1}{2}(a_i + a_{i+1})$. First, note that $\phi(x_A)$ must belong to an arc of $(g')^{-1}((a_i, a_{i+1}))$. Indeed, since $\|f - g' \circ \phi\|_\infty < (1 + 22K\varepsilon)$, one has $g'(\phi(x_A)) \in (\bar{a} - (1 + 22K)\varepsilon, \bar{a} + (1 + 22K)\varepsilon) \subseteq (a_i, a_{i+1})$. Then, according to the pigeonhole principle, there exist $x_A, x_{A'}$ such that $\phi(x_A)$ and $\phi(x_{A'})$ belong to the same arc of $(g')^{-1}((a_i, a_{i+1}))$.

- Since x_A and $x_{A'}$ do not belong to the same arc, we have

$$d_f(x_A, x_{A'}) > a_f/2.$$

- Now, since $\|f - g' \circ \phi\|_\infty < (1+22K)\varepsilon$ and $\phi(x_A), \phi(x_{A'})$ belong to the same arc of $(g')^{-1}((a_i, a_{i+1}))$, we also have (see Figure 3.9 for an illustration):

$$d_{g'}(\phi(x_A), \phi(x_{A'})) < 2(1 + 22K)\varepsilon.$$

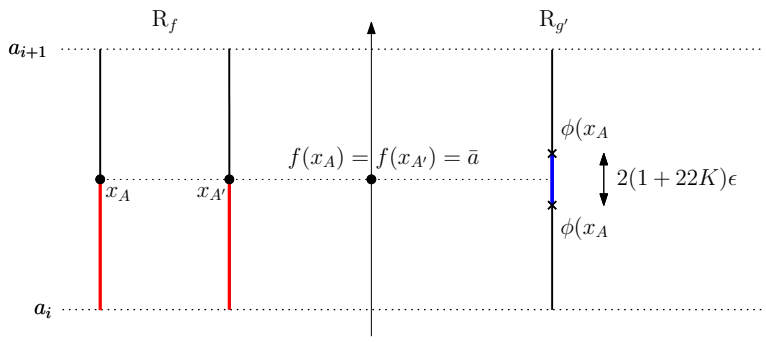

Figure 3.9: Any path between x_A and $x_{A'}$ must contain the red segments, and the blue segment is a particular path between $\phi(x_A)$ and $\phi(x_{A'})$.

Hence, $D(\phi, \psi) \geq |d_f(x_A, x_{A'}) - d_{g'}(\phi(x_A), \phi(x_{A'}))| > a_f/2 - 2(1+22K)\varepsilon$, which is greater than $2(1+22K)\varepsilon$ because $\varepsilon < a_f/(8(1 + 22K))$. Thus, $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) > (1 + 22K)\varepsilon$, which leads to a contradiction with (3.5). This means that there cannot be more arcs in $f^{-1}((a_i, a_{i+1}))$ than in $(g')^{-1}((a_i, a_{i+1}))$. Since the proof is symmetric in f and g', the numbers of arcs in $(g')^{-1}((a_i, a_{i+1}))$ and in $f^{-1}((a_i, a_{i+1}))$ are actually the same.

Finally, we show that the attaching maps of these arcs are also the same. In this particular graph setting, this is equivalent to showing that corresponding arcs in R_f and $\mathrm{R}_{g'}$ have the same endpoints. Let a_i be a critical value. Let $A^-_{f,i}$ and $A^+_{f,i}$ (resp. $A^-_{g',i}$ and $A^+_{g',i}$) be the sets of arcs in $f^{-1}((a_{i-1}, a_i))$ and $f^{-1}((a_i, a_{i+1}))$ (resp. $(g')^{-1}((a_{i-1}, a_i))$ and $(g')^{-1}((a_i, a_{i+1}))$). Morevover, we let ζ^i_f and ξ^i_f (resp. $\zeta^i_{g'}$ and $\xi^i_{g'}$) be the corresponding attaching maps that send arcs to their endpoints in $f^{-1}(a_i)$ (resp.

$(g')^{-1}(a_i))$. Let $A, B \in A^-_{f,i}$. We define an equivalence relation $\sim_{f,i}$ between A and B by: $A \sim_{f,i} B$ if and only if $\zeta^i_f(A) = \zeta^i_f(B)$, i.e. the endpoints of the arcs in the critical slice $f^{-1}(a_i)$ are the same. Similarly, $C, D \in A^+_{f,i}$ are equivalent if and only if $\xi^i_f(C) = \xi^i_f(D)$. One can define $\sim_{g',i}$ in the same way. To show that the attaching maps of R_f and $\mathrm{R}_{g'}$ are the same, we need to find a bijection b between the arcs of R_f and $\mathrm{R}_{g'}$ such that $A \sim_{f,i} B \Leftrightarrow b(A) \sim_{g',i} b(B)$ for each i.

We will now define b then check that it satisfies the condition. Recall from (3.5) that $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) < (1 + 22K)\varepsilon$. Hence there exists a continuous map $\phi : \mathrm{R}_f \to \mathrm{R}_{g'}$ such that $\|f - g' \circ \phi\|_\infty < (1 + 22K)\varepsilon$. This map induces a bijection b between the arcs of R_f and $\mathrm{R}_{g'}$. Indeed, given an arc $A \in A^-_{f,i}$, let $x \in A$ such that $f(x) = \bar{a} = \frac{1}{2}(a_{i-1} + a_i)$. We define $b(A)$ as the arc of $A^-_{g,i}$ that contains $\phi(x)$. The map b is well-defined since $g' \circ \phi(x) \in [\bar{a} - (1 + 22K)\varepsilon, \bar{a} + (1 + 22K)\varepsilon] \subseteq (a_{i-1}, a_i)$, hence $\phi(x)$ must belong to an arc of $(g')^{-1}((a_{i-1}, a_i))$. Let us show that $b(A) \sim_{g',i} b(B) \Rightarrow A \sim_{f,i} B$. Assume there exist $A, B \in A^-_{f,i}$ (the treatment of $A, B \in A^+_{f,i}$ is similar) such that $A \not\sim_{f,i} B$ and $b(A) \sim_{g',i} b(B)$. Let $x = \zeta^i_f(A)$ and $y = \zeta^i_f(B)$. Then we have $d_f(x, y) \geq a_f$ while $d_{g'}(\phi(x), \phi(y)) < 2(1 + 22K)\varepsilon$ (see Figure 3.10). Hence $|d_f(x, y) - d_{g'}(\phi(x), \phi(y))| > a_f - 2(1+22K)\varepsilon > 2(1+22K)\varepsilon$, so $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_{g'}) > (1+22K)\varepsilon$, which leads to a contradiction with (3.5). The same argument applies to show that $A \sim_{f,i} B \Rightarrow b(A) \sim_{g',i} b(B)$. □

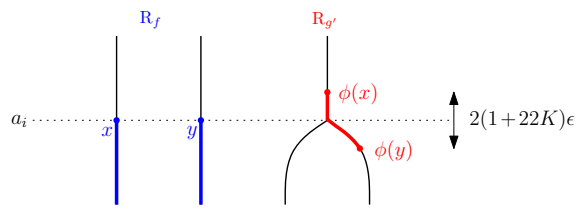

Figure 3.10: Any path from x to y must go through an entire arc, hence $d_f(x, y) \geq a_f$. On the contrary, there exists a direct path (displayed in red) between $\phi(x)$ and $\phi(y)$, hence $d_{g'}(\phi(x), \phi(y)) < 2(1+22K)\varepsilon$.

Hence, d_{b} and d_{FD} are locally equivalent, and so are d_{b} and d_{fGH} thanks to Theorem 2.4.9.

3.3 Induced Metrics

A desired property for dissimilarity measures is to be *intrinsic*, i.e. realized as the lengths of shortest continuous paths in the space of Reeb graphs [22]. This is particularly useful when one actually needs to interpolate between data, and not just discriminate between them, which happens in applications such as image or 3D shape morphing, skeletonization, and matching [74, 101, 104, 135]. Unfortunately, all the metrics proposed so far for Reeb graphs fail according to this criterion. Defining intrinsic metrics would not only open the door to the use of Reeb graphs in the aforementioned applications, but it would also provide a better understanding of the intrinsic structure of the space of Reeb graphs, and give a deeper meaning to the distance values.

In this section, we leverage the local equivalence given by Theorem 3.0.1 to derive a global equivalence between the intrinsic metrics $\hat{d}_{\mathrm{b}}$ and $\hat{d}_{\mathrm{FD}}$ induced by d_{b} and d_{FD}. Note that we already know $\hat{d}_{\mathrm{FD}}$ to be equivalent to $\hat{d}_{\mathrm{fGH}}$ since d_{FD} is equivalent to d_{fGH}.

Notation. Let Reeb denote the space of Reeb graphs coming from Morse-type functions. In the following, whatever the metric $d : \mathsf{Reeb} \times \mathsf{Reeb} \to \mathbb{R}_+$ under consideration, we define the class of *admissible paths* in Reeb to be those maps $\gamma : [0, 1] \to \mathsf{Reeb}$ that are continuous in d_{FD}. This makes sense when d is either d_{FD} itself or d_{fGH}, which is equivalent to d_{FD} and therefore admits the same continuous maps $\gamma : [0, 1] \to \mathsf{Reeb}$. In the case $d = d_{\mathrm{b}}$ our convention means restricting the class of admissible paths to a strict subset of the maps $\gamma : [0, 1] \to \mathsf{Reeb}$ that are continuous in d_{b} (by Theorem 2.4.13), which is required by some of our following claims.

Definition 3.3.1. *Let $d : \mathsf{Reeb} \times \mathsf{Reeb} \to \mathbb{R}_+$ be a metric on Reeb. Let $\mathrm{R}_f, \mathrm{R}_g \in \mathsf{Reeb}$, and $\gamma : [0, 1] \to \mathsf{Reeb}$ be an admissible path such that $\gamma(0) = \mathrm{R}_f$ and $\gamma(1) = \mathrm{R}_g$. The* length *of γ induced by d is defined as $L_d(\gamma) = \sup_{n,\Sigma} \sum_{i=0}^{n-1} d(\gamma(t_i), \gamma(t_{i+1}))$ where n ranges over $\mathbb{N}$ and Σ ranges over all*

partitions $0 = t_0 \leq t_1 \leq ... \leq t_n = 1$ *of* $[0,1]$. *The* intrinsic metric induced by d, *denoted* $\hat{d}$, *is defined by* $\hat{d}(\mathrm{R}_f, \mathrm{R}_g) = \inf_\gamma \mathrm{L_d}(\gamma)$ *where* γ *ranges over all admissible paths* $\gamma : [0,1] \to \mathsf{Reeb}$ *such that* $\gamma(0) = \mathrm{R}_f$ *and* $\gamma(1) = \mathrm{R}_g$.

Strong equivalence of induced metrics. The following result is, in our view, the starting point for the study of intrinsic metrics over the space of Reeb graphs. It comes as a consequence of the (local or global) equivalences between d_b and d_FD stated in Theorems 2.4.13 and 3.0.1. The intuition is that integrating two locally equivalent metrics along the same path using sufficiently small integration steps yields the same total length up to a constant factor, hence the global equivalence between the induced intrinsic metrics[1].

Theorem 3.3.2. $\hat{d}_\mathrm{b}$ *and* $\hat{d}_\mathrm{FD}$ *are globally equivalent. Specifically, for any* $\mathrm{R}_f, \mathrm{R}_g \in \mathsf{Reeb}$,

$$\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g)/22 \leq \hat{d}_\mathrm{b}(\mathrm{R}_f, \mathrm{R}_g) \leq 2\,\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g). \tag{3.6}$$

Proof. We first show that $\hat{d}_\mathrm{b}(\mathrm{R}_f, \mathrm{R}_g) \leq 2\,\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g)$. Let γ be an admissible path and let $\Sigma = \{t_0, ..., t_n\}$ be a partition of $[0,1]$. Then, by Theorem 2.4.13,

$$\sum_{i=0}^{n-1} d_\mathrm{FD}(\gamma(t_i), \gamma(t_{i+1})) \geq \frac{1}{2} \sum_{i=0}^{n-1} d_\mathrm{b}(\gamma(t_i), \gamma(t_{i+1})).$$

Since this is true for any partition Σ of any finite size n, it follows that

$$L_{d_\mathrm{FD}}(\gamma) \geq \frac{1}{2} L_{d_\mathrm{b}}(\gamma) \geq \frac{1}{2} \hat{d}_\mathrm{b}(\mathrm{R}_f, \mathrm{R}_g).$$

Again, this inequality holds for any admissible path γ, so $\hat{d}_\mathrm{b}(\mathrm{R}_f, \mathrm{R}_g) \leq 2\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g)$.
We now show that $\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g)/22 \leq \hat{d}_\mathrm{b}(\mathrm{R}_f, \mathrm{R}_g)$. Let γ be an admissible path and $\Sigma = \{t_0, ..., t_n\}$ a partition of $[0,1]$. We claim that there is a refinement of Σ (i.e. a partition $\Sigma' = \{t'_0, ..., t'_m\} \supseteq \Sigma$ for some $m \geq n$) such that $d_\mathrm{FD}(\gamma(t'_j), \gamma(t'_{j+1})) < \max\{a_{t'_j}, a_{t'_{j+1}}\}/16$ for all $j \in \{0, ..., m-1\}$, where $a_t > 0$ denotes the minimal distance between consecutive critical values of $\gamma(t)$. Indeed, since γ is continuous in d_FD, for any $t \in [0,1]$ there exists $\delta_t > 0$ such that $d_\mathrm{FD}(\gamma(t), \gamma(t')) < a_t/16$ for all $t' \in [0,1]$ with $|t - t'| < \delta_t$. Consider the open cover $\{(\max\{0, t - \delta_t/2\}, \min\{1, t + \delta_t/2\})\}_{t \in [0,1]}$ of $[0,1]$. Since $[0,1]$ is compact, there exists a finite subcover containing all the intervals $(t_i - \delta_{t_i}/2, t_i + \delta_{t_i}/2)$ for $t_i \in \Sigma$. Assume without loss of generality that this subcover is minimal (if it is not, then reduce the δ_{t_i} as much as needed). Let then $\Sigma' = \{t'_0, ..., t'_m\} \supseteq \Sigma$ be the partition of $[0,1]$ given by the midpoints of the intervals in this subcover, sorted by increasing order. Since the subcover is minimal, we have $t'_{j+1} - t'_j < (\delta_{t'_j} + \delta_{t'_{j+1}})/2 < \max\{\delta_{t'_j}, \delta_{t'_{j+1}}\}$ hence $d_\mathrm{FD}(\gamma(t'_j), \gamma(t'_{j+1})) < \max\{a_{t'_j}, a_{t'_{j+1}}\}/16$ for each $j \in \{0, m-1\}$. It follows that

$$\begin{aligned}
\sum_{i=0}^{n-1} d_\mathrm{FD}(\gamma(t_i), \gamma(t_{i+1})) &\leq \sum_{j=0}^{m-1} d_\mathrm{FD}(\gamma(t'_j), \gamma(t'_{j+1})) \text{ by the triangle inequality since } \Sigma' \supseteq \Sigma \\
&\leq 22 \sum_{j=0}^{m-1} d_\mathrm{b}(\gamma(t'_j), \gamma(t'_{j+1})) \text{ by Theorem 3.0.1 with } K = 1/22 \\
&\leq 22\, L_{d_\mathrm{b}}(\gamma).
\end{aligned}$$

Since this is true for any partition Σ of any finite size n, it follows that

$$\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g) \leq L_{d_\mathrm{FD}}(\gamma) \leq 22\, L_{d_\mathrm{b}}(\gamma).$$

Again, this inequality is true for any admissible path γ, so $\hat{d}_\mathrm{FD}(\mathrm{R}_f, \mathrm{R}_g) \leq 22\,\hat{d}_\mathrm{b}(\mathrm{R}_f, \mathrm{R}_g)$. □

[1]Provided the induced metrics are defined using the same class of admissible paths, hence our convention.

Consequences of the strong equivalence. Theorem 3.3.2 implies in particular that $\hat{d}_{\mathrm{b}}$ is a true metric on Reeb graphs, as opposed to d_{b} which is only a pseudometric. Moreover, the simplification operator defined in Section 3.2 makes it possible to continuously deform any Reeb graph into a trivial segment-shaped graph then into the empty graph. This shows that Reeb is path-connected in d_{FD}. Since the length of such continuous deformations is finite if the Reeb graph is finite, $\hat{d}_{\mathrm{FD}}$ and $\hat{d}_{\mathrm{b}}$ are finite metrics. Finally, the global equivalence of $\hat{d}_{\mathrm{FD}}$ and $\hat{d}_{\mathrm{b}}$ yields the following:

Corollary 3.3.3. *The metrics $\hat{d}_{\mathrm{FD}}$ and $\hat{d}_{\mathrm{b}}$ induce the same topology on* Reeb*, which is a refinement of the ones induced by d_{FD} or d_{b}.*

Note that the first inequality in (3.6) and, consequently, Corollary 3.3.3, are wrong if one defines the admissible paths for $\hat{d}_{\mathrm{b}}$ to be the whole class of maps $[0,1] \to$ Reeb that are continuous in d_{b}—hence our convention. For instance, let us consider the two Reeb graphs R_f and R_g of Figure 2.14 such that $\mathrm{ExDg}(f) = \mathrm{ExDg}(g)$, and let us define $\gamma : [0,1] \to$ Reeb by $\gamma(t) = \mathrm{R}_f$ if $t \in [0, 1/2)$ and $\gamma(t) = \mathrm{R}_g$ if $t \in [1/2, 1]$. Then γ is continuous in d_{b} while it is not in d_{FD} at $1/2$ since $d_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_g) > 0$. In this case, $\hat{d}_{\mathrm{b}}(\mathrm{R}_f, \mathrm{R}_g) \leq L_{d_{\mathrm{b}}}(\gamma) = 0 < \hat{d}_{\mathrm{FD}}(\mathrm{R}_f, \mathrm{R}_g)$.

3.4 Conclusion

In this chapter, we proved that the bottleneck distance, even though it is only a pseudometric on Reeb graphs, can actually discriminate a Reeb graph from the other Reeb graphs in a small enough neighborhood, as efficiently as the other metrics do. This theoretical result legitimates the use of the bottleneck distance to discriminate between Reeb graphs in applications. It also motivates the study of intrinsic metrics, which can potentially shed new light on the structure of the space of Reeb graphs and open the door to new applications where interpolation plays a key part.

Among the future perspectives of this work are the following questions:

- **Can the lower bound be improved?** We believe that $\varepsilon/22$ is not optimal. Specifically, a more careful analysis of the simplification operator should allow one to derive a tighter upper bound than the one in Lemma 2.4.16, and to improve the current lower bound on d_{b}.
- **Do shortest paths exist in** Reeb**?** The existence of shortest paths achieving $\hat{d}_{\mathrm{b}}$ is an important question since a positive answer would enable one to define and study the *intrinsic curvature* of Reeb. Moreover, characterizing and computing these shortest paths would be useful for interpolating between Reeb graphs in applications. The existence of shortest paths is guaranteed e.g. when the space is complete and locally compact. Unfortunately, Reeb is not complete, as shown by the counter-example of Figure 3.11. A workaround would be to restrict the focus to the subspace of Reeb graphs having at most N features with height at most H, for fixed but arbitrary $N, H > 0$. We believe this subspace should be complete and locally compact, like its counterpart in the space of persistence diagrams [17].

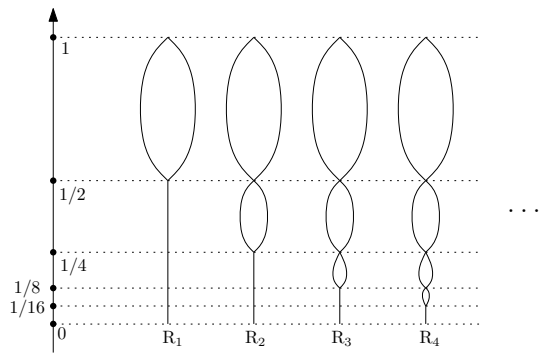

Figure 3.11: A sequence of Reeb graphs that is Cauchy but that does not converge in Reeb because the number of critical values goes to $+\infty$. Indeed, each R_n has $n+2$ critical values.

- **Is Reeb an Alexandrov space?** Provided shortest paths exist in Reeb (or in some subspace thereof), one can investigate whether the intrinsic curvature is bounded, either from above or from below. This is interesting because barycenters in metric spaces with bounded curvature enjoy many useful properties [111], and they can be approximated effectively in practice [110].
- **Can the local equivalence be extended to general metric spaces?** We have reasons to believe that our local equivalence result can be used to prove similar results for more general classes of metric spaces than Reeb graphs. If true, this would shed new light on inverse problems in persistence theory.

CHAPTER 4

STRUCTURE AND STABILITY OF THE MAPPER

In Chapter 3, we have seen how distances between Reeb graphs can be related to each other. In this chapter, we turn the focus on studying the structure and defining stable distances between Mappers, which are pixelized versions of Reeb graphs.

Indeed, somewhat surprisingly, despite its success in applications, very little is known to date about the structure of the Mapper and its stability with respect to perturbations of the data or of the cover. Intuitively, as a pixelized version of the Reeb graph, the Mapper should capture some of its topological features (branches, holes) and miss others, depending on how its cover is positioned. The stability of the structure of the Mapper, and thus the corresponding distance used to compare them, should also depend on this positioning.

The main result of this chapter is to formalize this intuition. We show in Theorem 4.3.3 that the topological structure of the Mapper can be read off from the one of the Reeb graph through a simple procedure. We build on this procedure to show Theorem 4.4.2, which states that Mappers are actually stable when compared with an appropriate distance. More precisely, we show that:

- $\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I})) = \mathrm{ExDg}(\mathrm{R}_f(X)) \setminus Q_{\mathcal{I}}$ is a bag-of-features signature of the topological structure of the Mapper, and can be computed solely by removing points of $\mathrm{ExDg}(\mathrm{R}_f(X))$ that belong to a specific area $Q_{\mathcal{I}}$ of the plane, which only depends on the cover $\mathcal{I}$ (Theorem 4.3.3),

- this signature is stable: $d_{\mathcal{I}}(\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I})), \mathrm{ExDg}(\mathrm{M}_g(X,\mathcal{I}))) \leq \|f-g\|_\infty$, where $d_{\mathcal{I}}$ is a distance depending only on $Q_{\mathcal{I}}$ (Theorem 4.4.2).

The area $Q_{\mathcal{I}}$ is a direct measure of the approximation quality of the Mapper: if $\mathcal{I}$ contains large intervals, then many points of $\mathrm{ExDg}(\mathrm{R}_f(X))$ will be included in $Q_{\mathcal{I}}$, and thus the Mapper is going to be a very rough approximation of the Reeb graph. This is formalized in Corollary 4.3.6.

We end the chapter by showing that any Mapper is actually isomorphic to a specific Reeb graph, whose connection to the one that the Mapper is approximating can be controlled in both the bottleneck (Theorem 4.6.12) and functional distortion distance (Theorem 4.6.10).

To prove all of these results, we use an intermediate construction called the *MultiNerve Mapper*, which is a slight, and somehow natural, extension of the usual Mapper.

Plan of the Chapter. We first give properties of Mappers computed with scalar-valued functions in Section 4.1. We then detail a variant therof, the *MultiNerve Mapper*, in Section 4.2. Next, we show how the topological structure of the (MultiNerve) Mapper can actually be derived from the one of the Reeb graph in Section 4.3. This allows us to define an adequate and computable pseudometric to compare the (MultiNerve) Mappers and provide stability results in Sections 4.4 and 4.5. Finally, we use the telescope operators of Section 3.1 to provide a convergence result of the (MultiNerve) Mapper to the Reeb graph in the functional distortion distance in Section 4.6.

4.1 Mappers for scalar-valued functions

We begin the chapter with some remarks on Mappers computed with scalar-valued functions. In particular, we show that, for specifics covers of the real line called *gomics*, these 1-dimensional Mappers have multigraph structures.

Interval cover. Let Z be a subset of $\mathbb{R}$, equipped with the subspace topology. A subset $U \subseteq Z$ is an *interval of* Z if there is an interval I of $\mathbb{R}$ such that $U = I \cap Z$. Note that U is open in Z if and only if I can be chosen open in $\mathbb{R}$. A cover $\mathcal{I}$ of Z is an *interval cover* if all its elements are intervals. In this case, $\mathrm{End}(\mathcal{I})$ denotes the set of all of the interval endpoints. Finally, the *granularity* of $\mathcal{I}$ is the supremum of the lengths of its elements, i.e. it is the quantity $\sup_{I \in \mathcal{I}} |I|$ where $|I| = \sup(I) - \inf(I) \in \mathbb{R} \cup \{+\infty\}$.

Lemma 4.1.1. *No more than two elements of an open minimal interval cover can intersect at a time.*

Proof. Assume for a contradiction that there are $k \geq 3$ elements of $\mathcal{I}$: $U_1, \cdots, U_k$, that have a non-empty common intersection. For every i, fix an open interval I_i of $\mathbb{R}$ such that $U_i = I_i \cap Z$. Up to a reordering of the indices, we can assume without loss of generality that I_1 has the smallest lower bound and I_2 has the largest upper bound. Since $I_1 \cap I_2 \supseteq U_1 \cap U_2 \neq \emptyset$, the remaining intervals satisfy $I_i \subseteq I_1 \cup I_2$. In particular, we have $U_3 = I_3 \cap Z \subseteq (I_1 \cup I_2) \cap Z = (I_1 \cap Z) \cup (I_2 \cap Z) = U_1 \cup U_2$, so the cover $\mathcal{I}$ is not minimal. □

Lemma 4.1.2. *If Z is $\mathbb{R}$ or a compact subset thereof, any cover $\mathcal{I}$ of Z has a minimal subcover.*

Proof. When Z is compact, there exists a subcover $\mathcal{J}$ of $\mathcal{I}$ that has finitely many elements. Any subcover of $\mathcal{J}$ with the minimum number of elements is then a minimal cover of Z.

When $Z = \mathbb{R}$, the same argument applies to any subset of the form $[-n, n]$, $n \in \mathbb{N}$. Then, a simple induction on n allows us to build a minimal subcover of $\mathcal{I}$. □

Gomics. From now on, unless otherwise stated, all covers of $Z \subseteq \mathbb{R}$ will be generic, open, minimal, interval covers (*gomics* for short). Given such a cover $\mathcal{I}$, the proper subset $\tilde{I}$ (as defined in Section 2.5) of any interval $I \in \mathcal{I}$ is itself an interval of Z since $\mathcal{I}$ is generic, therefore we call it the *proper subinterval* of I. Moreover, Lemma 4.1.1 yields a total order on the intervals of $\mathcal{I}$, so each one of them partitions into subintervals as follows:

$$I = I_{\cap}^{-} \sqcup \tilde{I} \sqcup I_{\cap}^{+}, \tag{4.1}$$

where $I_{\cap}^{-}$ is the intersection of I with the element right below it in the cover ($I_{\cap}^{-} = \emptyset$ if that element does not exist), and where $I_{\cap}^{+}$ is the intersection of I with the element right above it ($I_{\cap}^{+} = \emptyset$ if that element does not exist).

Mappers computed with gomics. Let X be a topological space and $f : X \to \mathbb{R}$ be a Morse-type function, whose image is covered by the cover $\mathcal{I}$. If $\mathcal{I}$ is a gomic, then the Mapper $\mathrm{M}_f(X, \mathcal{I})$ has a natural 1-dimensional stratification since no more than two intervals can intersect at a time by Lemma 4.1.1. Hence, in this case, it has the structure of a (possibly infinite) combinatorial graph and therefore has trivial homology in dimension 2 and above.

4.2 MultiNerve Mapper

In this section, we define a slight modification of the Mapper called the *MultiNerve Mapper*, which can be easily related to the Mapper—see Corollary 4.2.5, and whose analysis is more natural to handle.

Simplicial Posets. The MultiNerve Mapper construction is based on *multinerves*, which are specific *simplicial posets*.

Definition 4.2.1 ([56])**.** *A* simplicial poset *is a partially ordered set $(P, \preceq)$, whose elements are called* simplices*, such that:*

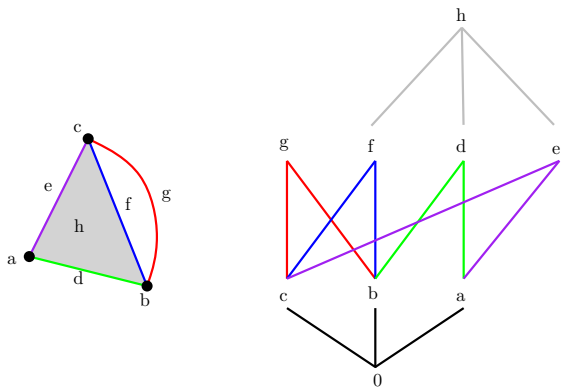

Figure 4.1: Left: A simplicial poset that is not a simplicial complex. Indeed, edges f and g have the same vertices (b and c). Right: The corresponding Hasse diagram showing the partial order on the simplices. Note that f, g cannot be part of the same 2-cell.

(i) *P has a least element called 0 such that $\forall p \in P$, $0 \preceq p$;*

(ii) *$\forall p \in P$, $\exists d \in \mathbb{N}$ such that the lower segment $[0, p] = \{q \in P : q \preceq p\}$ is* isomorphic *to the set of simplices of the standard d-simplex with the inclusion as partial order, where an isomorphism between posets is a bijective and order-preserving function.*

Simplicial posets are extensions of simplicial complexes: while every simplicial complex is also a simplicial poset (with inclusion as partial order and $\emptyset$ as least element), the converse is not always true as different simplices may have the same set of vertices. However, these simplices cannot be faces of the same higher-dimensional simplex, otherwise (ii) would be false. See Figure 4.1 for an example of a simplicial poset that is not a simplicial complex.

Multinerve. Given a cover $\mathcal{U}$ of X, the nerve is extended to a simplicial poset as follows:

Definition 4.2.2. *Let $\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$ be a cover of a topological space X. The* multinerve *$\mathcal{M}(\mathcal{U})$ is the simplicial poset defined by:*

$$\mathcal{M}(\mathcal{U}) = \left\{ (\{\alpha_0, \cdots, \alpha_k\}, C) : \bigcap_{i=0}^{k} U_{\alpha_i} \neq \emptyset \text{ and } C \text{ is a connected component of } \bigcap_{i=0}^{k} U_{\alpha_i} \right\}.$$

The proof that this set, together with the least element $(\emptyset, \bigcup_{\alpha \in A} U_\alpha)$ and the partial order $(F, C) \preceq (F', C') \Leftrightarrow F \subseteq F'$ and $C' \subseteq C$, is a simplicial poset, can be found in [56]. Given a simplex (F, C) in the multinerve of a cover, its *dimension* is $\mathrm{card}(F) - 1$. The dimension of the multinerve of a cover is the maximal dimension of its simplices. Given two simplices $(F, C), (F', C')$, the pair (F, C) is a *face* of (F', C') if $(F, C) \preceq (F', C')$.

MultiNerve Mapper. Given a connected pullback cover $\mathcal{V}$, we extend the Mapper by using the multinerve $\mathcal{M}(\mathcal{V})$ instead of $\mathcal{N}(\mathcal{V})$. This variant will be referred to as the MultiNerve Mapper in the following.

Definition 4.2.3. *Let X, Z be topological spaces, $f : X \to Z$ be a continuous function, $\mathcal{U}$ be a cover of $\mathrm{im}(f)$ and $\mathcal{V}$ be the associated connected pullback cover.*

Then, the MultiNerve Mapper *of X is $\overline{\mathrm{M}}_f(X, \mathcal{U}) = \mathcal{M}(\mathcal{V})$.*

See Figure 2.16 for an illustration. For the same reasons as Mapper, when $Z = \mathbb{R}$ and $\mathcal{I}$ is a gomic of $\mathrm{im}(f)$, the MultiNerve Mapper $\overline{\mathrm{M}}_f(X, \mathcal{I})$ is a (possibly infinite) combinatorial multigraph having trivial homology in dimension 2 and above. Contrarily to the Mapper, the MultiNerve Mapper also takes the connected components of the intersections into account in its construction. As we shall see in Section 4.3, it is able to capture the same features as the Mapper but with coarser gomics, and it is more naturally related to the Reeb graph.

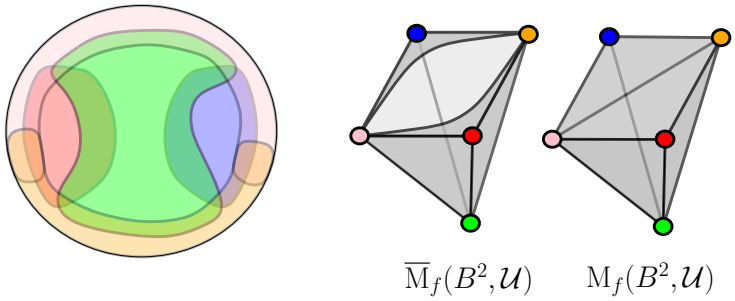

Figure 4.2: The domain is the disk B^2, and we consider the identity function f, as well as a generic open minimal cover $\mathcal{U}$ with five elements. The MultiNerve Mapper is homeomorphic to the disk B^2 and the Mapper is homeomorphic to the sphere $\mathbb{S}^2$. Then, $H_2(\mathrm{M}_f(B^2,\mathcal{U})) \neq 0$ while $H_2(\overline{\mathrm{M}}_f(B^2,\mathcal{U})) = 0$.

Connection to Mapper The connection between the Mapper and the MultiNerve Mapper is induced by the following connection between nerves and multinerves:

Lemma 4.2.4 ([56])**.** *Let X be a topological space and $\mathcal{U}$ be a cover of X. Let $\pi_1 : (F, C) \mapsto F$ be the projection of the simplices of $\mathcal{M}(\mathcal{U})$ onto the first coordinate. Then, $\pi_1(\mathcal{M}(\mathcal{U})) = \mathcal{N}(\mathcal{U})$.*

Corollary 4.2.5. *Let X, Z be topological spaces and let $f : X \to Z$ be a continuous function. Let $\mathcal{U}$ be a cover of $\mathrm{im}(f)$. Then, $\mathrm{M}_f(X,\mathcal{U}) = \pi_1(\overline{\mathrm{M}}_f(X,\mathcal{U}))$.*

Thus, when $Z = \mathbb{R}$ and $\mathcal{I}$ is a gomic, the Mapper $\mathrm{M}_f(X,\mathcal{I})$ is the simple graph obtained by gluing the edges that have the same endpoints in the MultiNerve Mapper. In this special case, it is even possible to embed $\mathrm{M}_f(X,\mathcal{I})$ as a subcomplex of $\overline{\mathrm{M}}_f(X,\mathcal{I})$. Indeed, both objects are multigraphs over the same set of nodes since they are built from the same connected pullback cover. Then, it is enough to map each edge of $\mathrm{M}_f(X,\mathcal{I})$ to one of its copies in $\overline{\mathrm{M}}_f(X,\mathcal{I})$, chosen arbitrarily, to get a subcomplex. This mapping serves as a simplicial section for the projection π_1, therefore:

Lemma 4.2.6. *When $Z = \mathbb{R}$ and $\mathcal{I}$ is a gomic, the projection π_1 defined in Lemma 4.2.4 induces a surjective homomorphism in homology.*

Note that this is not true in general when Z has a higher dimension—see Figure 4.2.

4.3 Structure of the MultiNerve Mapper

In this section, we study and characterize the topological structure of the (MultiNerve) Mapper computed on a non discrete topological space. More precisely, we show that this topological structure can be read off from the extended persistence diagram of the Reeb graph. To prove this, we show that the MultiNerve Mapper $\overline{\mathrm{M}}_f(X,\mathcal{I})$ is actually isomorphic (as a combinatorial multigraph) to a specific Reeb graph, whose extended persistence diagram is related to the extended persistence diagram $\mathrm{ExDg}(\tilde{f})$ of $\mathrm{R}_f(X)$.

Notation. In the following, the combinatorial version of the Reeb graph (where each critical point is turned into a node and where the functional and metric information is forgotten) is denoted by $\mathcal{C}\mathrm{R}_f(X)$.

4.3.1 Topological structure of the MultiNerve Mapper

In order to show that the MultiNerve Mapper is a specific Reeb graph, we first show that (MultiNerve) Mappers can be equipped with functions.

Definition 4.3.1. *Let $\mathcal{I} = \{I_\alpha\}_{\alpha\in A}$ be a gomic of* $\mathrm{im}(f)$ *and $\mathcal{V} = \{V_\alpha^i\}_{1\le i\le c(\alpha),\alpha\in A}$ be the associated connected pullback cover. Then we define $\bar{\mathsf{m}}_{\mathcal{I}} : \overline{\mathrm{M}}_f(X,\mathcal{I}) \to \mathbb{R}$ as the piecewise-linear extension of the function defined on the nodes of $\overline{\mathrm{M}}_f(X,\mathcal{I})$ by $V_\alpha^i \mapsto \mathrm{mid}(\tilde{I}_\alpha)$, where $\mathrm{mid}(\tilde{I}_\alpha)$ is the midpoint of the proper subinterval $\tilde{I}_\alpha$ of I_α. The definition of $\mathsf{m}_{\mathcal{I}} : \mathrm{M}_f(X,\mathcal{I}) \to \mathbb{R}$ is similar.*

Hence, Reeb graphs can be computed from $\overline{\mathrm{M}}_f(X,\mathcal{I})$ and $\mathrm{M}_f(X,\mathcal{I})$, once they are equipped with $\bar{\mathsf{m}}_{\mathcal{I}}$ and $\mathsf{m}_{\mathcal{I}}$ respectively. Let us call them $\mathrm{R}_{\bar{\mathsf{m}}_{\mathcal{I}}}(\overline{\mathrm{M}}_f(X,\mathcal{I}))$ and $\mathrm{R}_{\mathsf{m}_{\mathcal{I}}}(\mathrm{M}_f(X,\mathcal{I}))$, with corresponding induced maps $\tilde{\bar{\mathsf{m}}}_{\mathcal{I}} : \mathrm{R}_{\bar{\mathsf{m}}_{\mathcal{I}}}(\overline{\mathrm{M}}_f(X,\mathcal{I})) \to \mathbb{R}$ and $\tilde{\mathsf{m}}_{\mathcal{I}} : \mathrm{R}_{\mathsf{m}_{\mathcal{I}}}(\mathrm{M}_f(X,\mathcal{I})) \to \mathbb{R}$. The following lemma, which states that (MultiNerve) Mappers are isomorphic to their Reeb graphs, is a simple consequence of Remark 2.4.2.

Lemma 4.3.2. *Let X be a topological space and $f : X \to \mathbb{R}$ be a Morse-type function. Let $\mathcal{I}$ be a gomic of* $\mathrm{im}(f)$*. Then $\overline{\mathrm{M}}_f(X,\mathcal{I})$ and $\mathcal{C}\mathrm{R}_{\bar{\mathsf{m}}_{\mathcal{I}}}(\overline{\mathrm{M}}_f(X,\mathcal{I}))$ are isomorphic as combinatorial multigraphs. The same is true for $\mathrm{M}_f(X,\mathcal{I})$ and $\mathcal{C}\mathrm{R}_{\mathsf{m}_{\mathcal{I}}}(\mathrm{M}_f(X,\mathcal{I}))$.*

Hence, by a slight abuse of notation, we rename $\tilde{\mathsf{m}}_{\mathcal{I}}$ and $\tilde{\bar{\mathsf{m}}}_{\mathcal{I}}$ into $\mathsf{m}_{\mathcal{I}}$ and $\bar{\mathsf{m}}_{\mathcal{I}}$ for convenience.

We now state the main result of this section, which ensures that the extended persistence diagram $\mathrm{ExDg}(\bar{\mathsf{m}}_{\mathcal{I}})$, i.e. the bag-of-features signature of $\mathrm{R}_{\bar{\mathsf{m}}_{\mathcal{I}}}(\overline{\mathrm{M}}_f(X,\mathcal{I}))$ and $\overline{\mathrm{M}}_f(X,\mathcal{I})$, is nothing but a simplification of $\mathrm{ExDg}(\tilde{f})$, i.e. the bag-of-features signature of $\mathrm{R}_f(X)$.

Theorem 4.3.3. *Let X be a topological space and $f : X \to \mathbb{R}$ be a Morse-type function. Let $\mathrm{R}_f(X)$ be the corresponding Reeb graph and $\tilde{f} : \mathrm{R}_f(X) \to \mathbb{R}$ be the induced map. Let $\mathcal{I}$ be a gomic of* $\mathrm{im}(f)$*. There are bijections between:*

(i) $\mathrm{Ord}_0(\bar{\mathsf{m}}_{\mathcal{I}})$ *and* $\mathrm{Ord}_0(\tilde{f}) \setminus Q_O^{\mathcal{I}}$ (iii) $\mathrm{Ext}_1^-(\bar{\mathsf{m}}_{\mathcal{I}})$ *and* $\mathrm{Ext}_1^-(\tilde{f}) \setminus Q_{E^-}^{\mathcal{I}}$

(ii) $\mathrm{Rel}_1(\bar{\mathsf{m}}_{\mathcal{I}})$ *and* $\mathrm{Rel}_1(\tilde{f}) \setminus Q_R^{\mathcal{I}}$ (iv) $\mathrm{Ext}_0^+(\bar{\mathsf{m}}_{\mathcal{I}})$ *and* $\mathrm{Ext}_0^+(\tilde{f})$

where $Q_O^{\mathcal{I}} = \bigcup_{I\in\mathcal{I}} Q^+_{\tilde{I}\cup I_\cap^+}$, $Q_R^{\mathcal{I}} = \bigcup_{I\in\mathcal{I}} Q^-_{\tilde{I}\cup I_\cap^-}$, and $Q_{E^-}^{\mathcal{I}} = \bigcup_{I\in\mathcal{I}} Q_I^-$, and where, for any interval I with endpoints $a \le b$, we let $Q_I^+ = \{(x,y) \in \mathbb{R}^2 : a \le x \le y \le b\}$ be the corresponding half-square above the diagonal, and $Q_I^- = \{(x,y) \in \mathbb{R}^2 : a \le y < x \le b\}$ be the half-square strictly below the diagonal. See Figure 4.3 for an illustration.

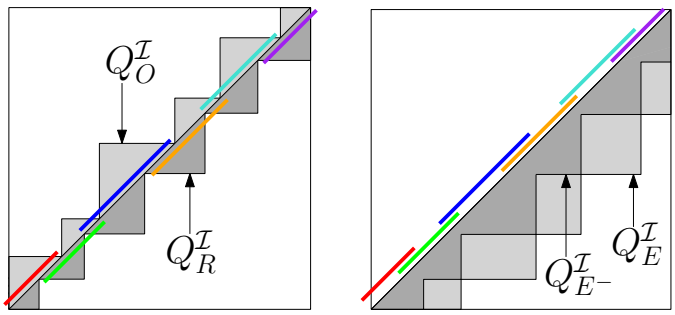

Figure 4.3: Left: Staircases of ordinary (light grey) and relative (dark grey) types. Right: Staircases of extended types—$Q_{E^-}^{\mathcal{I}}$ is in dark grey while $Q_E^{\mathcal{I}}$ is the union of $Q_{E^-}^{\mathcal{I}}$ with the light grey area.

The remaining of Section 4.3.1 is devoted to the proof of Theorem 4.3.3. In order to state the proof, we first introduce *cover zigzag persistence modules.*

Definition 4.3.4. *Let X be a topological space and $f : X \to \mathbb{R}$ be a Morse-type function. Let $\mathcal{I} = \{I_\alpha\}_{1\le\alpha\le m}$ be a gomic of* $\mathrm{im}(f)$*, sorted by the natural order defined in Section 4.1.*

Let $\mathrm{Crit}(f) = \{-\infty = a_0, a_1, ..., a_n, a_{n+1} = +\infty\}$*. For any open interval I with left endpoint a, we define the integers $l(I)$, $r(I)$ by $l(I) = \max\{i : a_i \le a\}$ and $r(I) = \max\{l(I), \max\{i : a_i \in I\}\}$. Then, we define the* cover zigzag persistence module $\mathrm{CZZ}(f,\mathcal{I})$ *by*

$$\mathrm{CZZ}(f,\mathcal{I}) = H_*\left(X_{l(I_1)}^{r(I_1)} \hookleftarrow X_{l(I_1\cap I_2)}^{r(I_1\cap I_2)} \hookrightarrow X_{l(I_2)}^{r(I_2)} \hookleftarrow X_{l(I_2\cap I_3)}^{r(I_2\cap I_3)} \hookrightarrow \cdots \hookleftarrow X_{l(I_{m-1}\cap I_m)}^{r(I_{m-1}\cap I_m)} \hookrightarrow X_{l(I_m)}^{r(I_m)}\right),$$

where the X_i^j spaces are as in Definition 2.3.5. We also let $\mathrm{CBc}(f,\mathcal{I})$ *denote the barcode of this module.*

Note that cover zigzag persistence modules can be isometrically embedded (with the bottleneck and Wasserstein distances) into the south face of the Mayer-Vietoris half-pyramid. Indeed, each node of $\mathrm{CZZ}(f,\mathcal{I})$ belongs to this south face. The only difficulty is that $\mathrm{CZZ}(f,\mathcal{I})$ may include the same node several times consecutively when there is a sequence of consecutive intervals in the gomic that are all included between two consecutive critical values of f, i.e. for which $l(I) = r(I)$. However, in that case, the corresponding arrows in the module are isomorphisms. Thus, composing these arrows leaves the resulting barcode unchanged.

Lemma 4.3.5. *Let X be a topological space and $f : X \to \mathbb{R}$ be a Morse-type function. Let $\mathcal{I}$ be a gomic of* $\mathrm{im}(f)$*. Then, there is a bijection between* $\mathrm{ExDg}(\bar{\mathsf{m}}_{\mathcal{I}})$ *and* $\mathrm{CBc}_0(f,\mathcal{I})$*.*

Proof. Recall from Corollary 2.3.8 that it suffices to show that $\mathrm{LZZ}_0(\bar{\mathsf{m}}_{\mathcal{I}})$ and $\mathrm{CZZ}_0(f,\mathcal{I})$ are isomorphic as zigzag persistence modules. Assume without loss of generality that $\mathcal{I}$ has m elements, with $m \in \mathbb{N}^*$. First, note that $\mathrm{card}(\mathrm{Crit}(\bar{\mathsf{m}}_{\mathcal{I}}))$ is equal to m. Hence, both $\mathrm{LZZ}(\bar{\mathsf{m}}_{\mathcal{I}})$ and $\mathrm{CZZ}(f,\mathcal{I})$ have exactly $2m+1$ nodes. Moreover, since the MultiNerve Mapper tracks the connected components of the interval and intersection preimages of f, each element of $\mathrm{LZZ}_0(\bar{\mathsf{m}}_{\mathcal{I}})$ is of the form $H_0(f^{-1}(I))$, $I \in \mathcal{I}$, or $H_0(f^{-1}(I \cap J))$, I, J consecutive in $\mathcal{I}$.

Let $I \in \mathcal{I}$. Since f is Morse-type, $X_{l(I)}^{r(I)}$ and $X^I = f^{-1}(I)$ have the same homotopy type. Indeed, recall from Definition 2.3.5 that there exist $s_{l(I)}$ and $s_{r(I)}$ such that $X_{l(I)}^{r(I)} = f^{-1}\left(\left[s_{l(I)}, s_{r(I)}\right]\right)$ and $s_{l(I)}$ (resp. $s_{r(I)}$) and the left (resp. right) endpoint of I are located between the same consecutive critical values of f. In particular, $X_{l(I)}^{r(I)}$ and X^I have the same number of connected components, meaning that $H_0(X^I)$ and $H_0(X_{l(I)}^{r(I)})$ are isomorphic groups. The same is also true for any $I \cap J$, $I, J \in \mathcal{I}$.

Hence, we define a canonical pointwise isomorphism Ψ in dimension 0 as follows: for each node, send each connected component of one preimage, or equivalently each generator of one homology group, to the connected component of the other preimage which intersects it (there is only one since the preimages have the same number of connected components). By definition of the MultiNerve Mapper, Ψ commutes with the canonical inclusion. Hence, $\mathrm{LZZ}_0(\bar{\mathsf{m}}_{\mathcal{I}})$ and $\mathrm{CZZ}_0(f,\mathcal{I})$ are isomorphic. □

Finally, we relate the cover zigzag persistence barcode to the extended persistence diagram of the Reeb graph. Namely, we show that a specific simplification of this extended persistence diagram encodes the same information as the cover zigzag persistence barcode.

Proof of Theorem 4.3.3. Again, recall from Corollary 2.3.8 that $\mathrm{ExDg}(\tilde{f})$ encodes the same information as $\mathrm{LBc}_0(\tilde{f})$. Hence, since $\mathrm{ExDg}(\bar{\mathsf{m}}_{\mathcal{I}})$ and $\mathrm{CBc}_0(f,\mathcal{I})$ are equivalent from Lemma 4.3.5, we focus on the relation between $\mathrm{LBc}_0(\tilde{f})$ and $\mathrm{CBc}_0(f,\mathcal{I})$. As mentioned after Definition 4.3.4, the cover zigzag persistence module $\mathrm{CZZ}(f,\mathcal{I})$ can be isometrically embedded in the south face of the Mayer-Vietoris half-pyramid. Hence, we can assume without loss of generality that the set of nodes of $\mathrm{CZZ}(f,\mathcal{I})$ is a subset of the nodes of a monotone zigzag module $\overline{\mathrm{CZZ}}(f,\mathcal{I})$ that can be drawn along the south face of the Mayer-Vietoris half-pyramid by interpolating the elements of $\mathrm{CZZ}(f,\mathcal{I})$. Thus, it suffices by Theorem 2.3.7 to study which intervals disappear when going from $\mathrm{LBc}_0(\tilde{f})$ to $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$ and then to $\mathrm{CBc}_0(f,\mathcal{I})$ using the pyramid rules recalled in Figure 4.4.

We first give analogues of staircases for zigzag persistence. For any $I = I_\cap^- \sqcup \tilde{I} \sqcup I_\cap^+ \in \mathcal{I}$, we define:

- $\mathrm{supp}_O(I)$ as the set of nodes of $\mathrm{LZZ}(f)$ that are located strictly between $X_{l(\tilde{I}\cup I_\cap^+)}^{l(\tilde{I}\cup I_\cap^+)}$ and $X_{r(\tilde{I}\cup I_\cap^+)-1}^{r(\tilde{I}\cup I_\cap^+)}$,
- $\mathrm{supp}_R(I)$ as the set of nodes of $\mathrm{LZZ}(f)$ that are located strictly between $X_{l(I_\cap^-\cup\tilde{I})}^{l(I_\cap^-\cup\tilde{I})+1}$ and $X_{r(I_\cap^-\cup\tilde{I})}^{r(I_\cap^-\cup\tilde{I})}$,
- $\mathrm{supp}_{E^-}(I)$ as the set of nodes of $\mathrm{LZZ}(f)$ that are located strictly between $X_{l(I)}^{l(I)+1}$ and $X_{r(I)-1}^{r(I)}$.

There are two possible ways for an interval of $\mathrm{LBc}_0(f)$ to disappear in $\mathrm{CBc}_0(f,\mathcal{I})$: either its homological dimension is shifted by 1, or its intersection with the set of nodes of $\mathrm{CZZ}(f,\mathcal{I})$ is empty after being projected onto $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$—see Figure 4.5. According to the pyramid rules, we have that:

- Projections of type III intervals of $\mathrm{LBc}_0(f)$ onto $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$ always intersect with the nodes of $\mathrm{CZZ}(f,\mathcal{I})$ and their homological dimensions cannot be shifted. Hence, none of them disappears. This proves (iv).

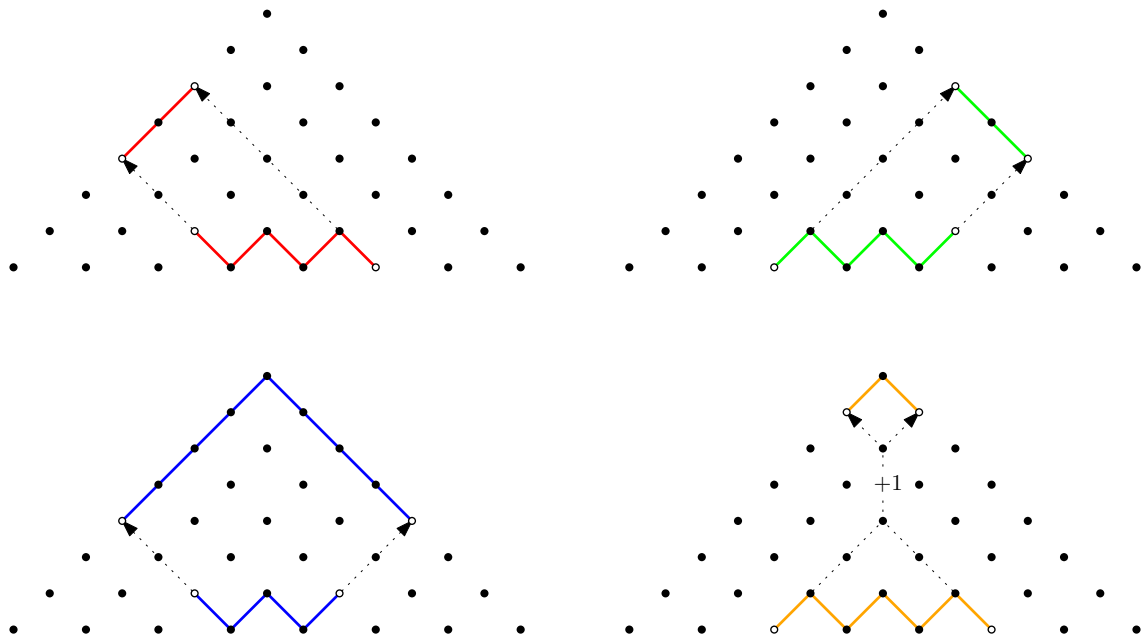

Figure 4.4: (From [28]) We show the axis of travel of birth and death endpoints of intervals of $\mathrm{LZZ}(f)$ to the up-down zigzag persistence module bounding the south face of the Mayer-Vietoris half-pyramid for interval modules that correspond to type I intervals (upper-left, red), type II intervals (upper-right, green), type III intervals (down-left, blue), and type IV intervals (down-right, orange) in $\mathrm{LBc}(f)$. The $+1$ in the down-right figure means that the homological dimension is increased by one.

- Projections of type IV intervals of $\mathrm{LBc}_0(f)$ onto $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$ always intersect with the nodes of $\mathrm{CZZ}(f,\mathcal{I})$. However, their homological dimensions can be shifted by 1. This happens when the endpoints collide in the south face of the Mayer-Vietoris half-pyramid. Hence, only those intervals whose support is included in $\mathrm{supp}_{E^-}(I)$ for some $I \in \mathcal{I}$ go through such a shift before getting to $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$. This proves (iii).
- Homological dimensions of type I intervals in $\mathrm{LBc}_0(f)$ cannot be shifted, but their projections onto $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$ may not always intersect with the nodes of $\mathrm{CZZ}(f,\mathcal{I})$. This happens for those intervals whose support is included in $\mathrm{supp}_O(I)$ for some $I \in \mathcal{I}$, thus proving (i).
- Homological dimensions of type II intervals in $\mathrm{LBc}_0(f)$ cannot be shifted, but their projections onto $\overline{\mathrm{CBc}}_0(f,\mathcal{I})$ may not always intersect with the nodes of $\mathrm{CZZ}(f,\mathcal{I})$. This happens for those intervals whose support is included in $\mathrm{supp}_R(I)$ for some $I \in \mathcal{I}$, thus proving (ii).

□

4.3.2 A signature for MultiNerve Mapper

Theorem 4.3.3 means that the dictionary introduced in Section 2.4.1 can be used to describe the structure of the MultiNerve Mapper from the extended persistence diagram of the induced function $\tilde{f}$. Indeed, the topological features of $\overline{\mathrm{M}}_f(X,\mathcal{I})$ are in bijection with the points of $\mathrm{ExDg}(\tilde{f})$ minus the ones that fall into the various staircases $(Q_O^{\mathcal{I}}, Q_{E^-}^{\mathcal{I}}, Q_R^{\mathcal{I}})$ corresponding to their type. Moreover, by Theorem 2.4.4, $\mathrm{ExDg}(\tilde{f})$ itself is obtained from $\mathrm{ExDg}_0(f)$ and $\mathrm{ExDg}_1(f)$ by removing the points of $\mathrm{Ext}_1^+(f)$ and $\mathrm{Ord}_1(f)$. Hence, we use the off-staircase part of $\mathrm{ExDg}(\tilde{f})$ as a signature for the structure of the MultiNerve Mapper[1]:

$$\begin{aligned}\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})) &= (\mathrm{Ord}(\tilde{f}) \setminus Q_O^{\mathcal{I}}) \cup (\mathrm{Ext}(\tilde{f}) \setminus Q_{E^-}^{\mathcal{I}}) \cup (\mathrm{Rel}(\tilde{f}) \setminus Q_R^{\mathcal{I}}) \\ &= (\mathrm{Ord}_0(f) \setminus Q_O^{\mathcal{I}}) \cup ((\mathrm{Ext}_0^+(f) \cup \mathrm{Ext}_1^-(f)) \setminus Q_{E^-}^{\mathcal{I}}) \cup (\mathrm{Rel}_1(f) \setminus Q_R^{\mathcal{I}}).\end{aligned} \tag{4.2}$$

[1]Recall that $\mathrm{Ext}_0^-(f) = \mathrm{Rel}_0(f) = \emptyset$.

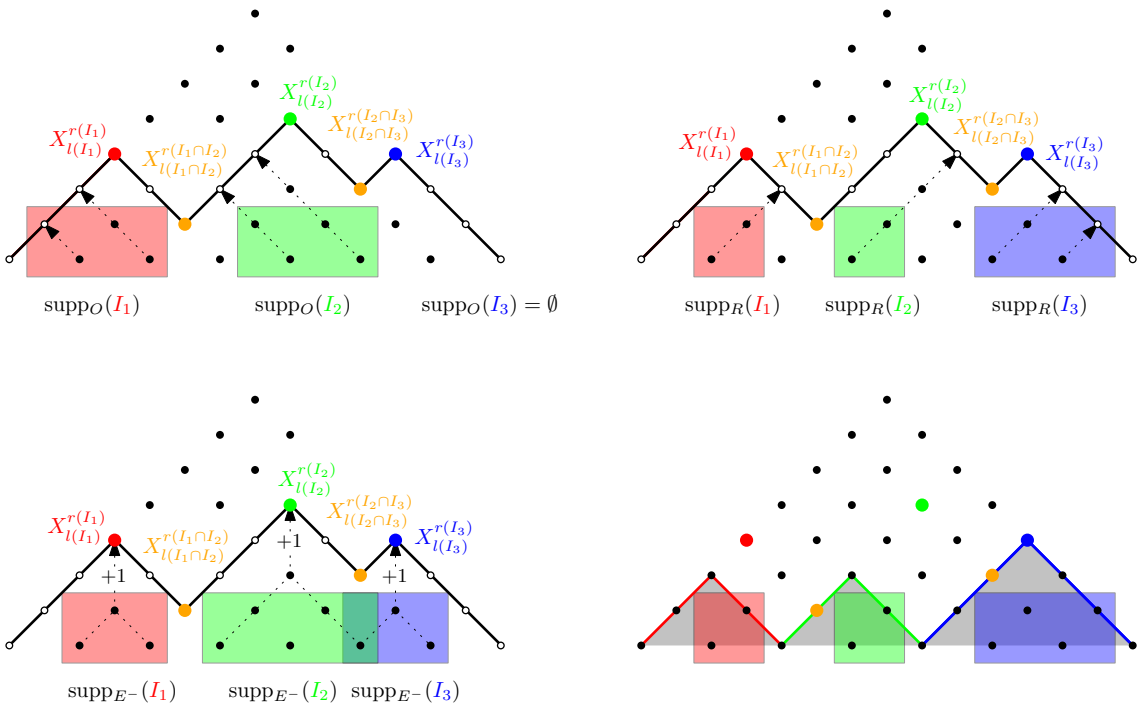

Figure 4.5: The black path in the south face of the Mayer-Vietoris half-pyramid represents the monotone zigzag persistence module $\overline{\mathrm{CZZ}}(f,\mathcal{I})$ for a gomic $\mathcal{I}$ with three intervals. The white disks on this path are the nodes that do not intersect the set of nodes of the cover zigzag persistence module $\mathrm{CZZ}(f,\mathcal{I})$, which are colored according to the interval of $\mathcal{I}$ they represent (and are colored orange if they represent an intersection). The boxes outline the support of the intervals of $\mathrm{LBc}_0(f)$ that disappear in the MultiNerve Mapper depending on their types (upper-left for type I intervals, upper-right for type II intervals and down-left for type IV intervals). We also show (down-right) the analogue, drawn in grey color, of $Q^{\mathcal{I}}_R$ on the south face of the Mayer-Vietoris half-pyramid.

We call this signature the *extended persistence diagram* of the MultiNerve Mapper. Note that this signature is not computed by applying persistence to some function defined on the multinerve, but it is rather a pruned version of the extended persistence diagram of $\tilde{f}$. As for Reeb graphs, it serves as a bag-of-features type signature of the structure of $\overline{\mathrm{M}}_f(X,\mathcal{I})$. Moreover, the fact that $\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})) \subseteq \mathrm{ExDg}(\tilde{f})$ formalizes the intuition that the MultiNerve Mapper should be viewed as a *pixelized version* of the Reeb graph, in which some of the features disappear due to the staircases (prescribed by the cover). For instance, in Figure 4.6 we show a double torus equipped with the height function, together with its associated Reeb graph, MultiNerve Mapper, and Mapper. We also show the corresponding extended persistence diagrams. In each case, the points in the diagram represent the features of the object: the extended points represent the holes (dimension 1 and above) and the trunks (dimension 0) while the ordinary and relative points represent the branches.

Convergence of the signature. The following convergence result (which is in fact non-asymptotic) is a direct consequence of our previous results:

Corollary 4.3.6. *Suppose the granularity of the gomic $\mathcal{I}$ is at most ε. Then,*

$$\mathrm{ExDg}(\tilde{f}) \setminus \{(x,y) \,:\, |y-x| \leq \varepsilon\} \subseteq \mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})) \subseteq \mathrm{ExDg}(\tilde{f}).$$

Thus, the features (branches, holes) of the Reeb graph that are missing in the MultiNerve Mapper have spans at most ε. In particular, we have $d_{\mathrm{b}}(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})), \mathrm{ExDg}(\tilde{f})) \leq \varepsilon/2$. Moreover, the two signatures become equal when ε becomes smaller than the smallest vertical distance of the points of $\mathrm{ExDg}(\tilde{f})$ to the diagonal. Finally, $\overline{\mathrm{M}}_f(X,\mathcal{I})$ and $\mathrm{R}_f(X)$ themselves become isomorphic as combinatorial graphs up to one-step vertex splits and edge subdivisions (which are topologically trivial modifications) when ε becomes smaller than the smallest absolute difference between distinct critical values of f.

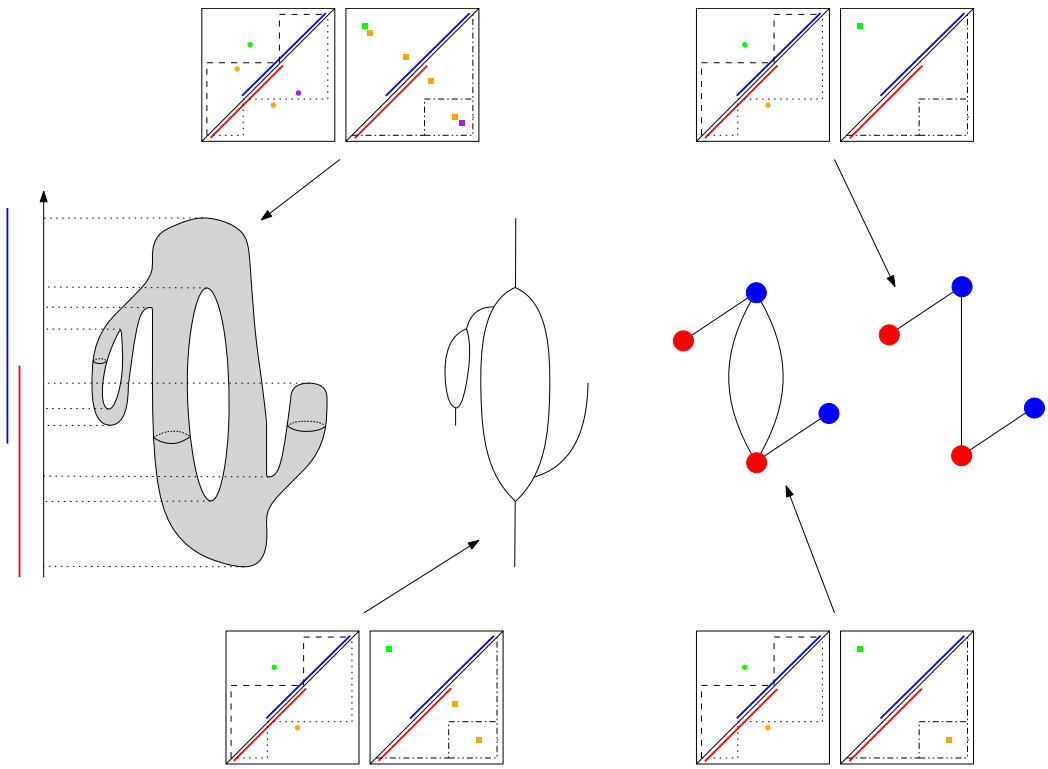

Figure 4.6: From left to right: a 2-manifold equipped with the height function; the corresponding Reeb graph, MultiNerve Mapper, and Mapper. For each object, we display the extended persistence diagrams of dimension 0 (green points), 1 (orange points) and 2 (purple points). Extended points are squares while ordinary and relative points are disks (above and below the diagonal respectively). The staircases are represented with dashed ($Q_O^{\mathcal{I}}$), dotted ($Q_{E^-}^{\mathcal{I}}$), dash-dotted ($Q_R^{\mathcal{I}}$), and dash-dot-dotted ($Q_E^{\mathcal{I}}$) lines. One can see how to go from the extended persistence diagram of the height function to the one of the induced map (remove the points in dimension 2 and the points in dimension 1 above the diagonal), then to the one of the MultiNerve Mapper (remove the points inside the staircases corresponding to their type), and finally, to the one of the Mapper (remove the extended points in $Q_E^{\mathcal{I}}$).

We show a similar convergence result in the functional distortion distance in Section 4.6. Note that building the signature $\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I}))$ requires computing the critical values of f exactly, which may not always be possible. However, as for Reeb graphs, the signature can be approximated efficiently and with theoretical guarantees under mild sampling conditions using existing work on scalar fields analysis, as we will see in Chapter 5.

4.3.3 Induced signature for Mapper

Recall from Lemma 4.2.6 that the projection $\pi_1 : \overline{\mathrm{M}}_f(X,\mathcal{I}) \to \mathrm{M}_f(X,\mathcal{I})$ induces a surjective homomorphism in homology. Thus, the Mapper has a simpler structure than the MultiNerve Mapper. To be more specific, π_1 identifies all the edges connecting the same pair of vertices. This eliminates the corresponding holes in $\overline{\mathrm{M}}_f(X,\mathcal{I})$. Since the two vertices lie in successive intervals of the cover, the corresponding diagram points lie in the following extended staircase (see the staircase $Q_E^{\mathcal{I}}$ displayed on the right in Figure 4.3):

$$Q_E^{\mathcal{I}} = \bigcup_{I\cup J \text{ such that } I\cap J\neq\emptyset} Q^-_{I\cup J}.$$

The other staircases remain unchanged. Hence the following signature:

$$\begin{aligned}\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I})) &= (\mathrm{Ord}(\tilde{f}) \setminus Q_O^{\mathcal{I}}) \cup (\mathrm{Ext}(\tilde{f}) \setminus Q_E^{\mathcal{I}}) \cup (\mathrm{Rel}(\tilde{f}) \setminus Q_R^{\mathcal{I}}) \\ &= (\mathrm{Ord}_0(f) \setminus Q_O^{\mathcal{I}}) \cup ((\mathrm{Ext}_0^+(f) \cup \mathrm{Ext}_1^-(f)) \setminus Q_E^{\mathcal{I}}) \cup (\mathrm{Rel}_1(f) \setminus Q_R^{\mathcal{I}}).\end{aligned} \tag{4.3}$$

The interpretation of this signature in terms of the structure of the Mapper follows the same rules as for the MultiNerve Mapper and Reeb graph—see again Figure 4.6. Moreover, the convergence result stated in Corollary 4.3.6 holds for the Mapper as well.

4.4 Stability in the bottleneck distance

Intuitively, for a point in the signature $\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I}))$, the ℓ^∞-distance to its corresponding staircase[2] measures the amount by which the function f or the cover $\mathcal{I}$ must be perturbed in order to eliminate the corresponding feature (branch, hole) in the MultiNerve Mapper. Conversely, for a point in the Reeb graph's signature $\mathrm{ExDg}(\tilde{f})$ that is not in the MultiNerve Mapper's signature (i.e. that lies inside its corresponding staircase), the ℓ^∞-distance to the boundary of the staircase measures the amount by which f or $\mathcal{I}$ must be perturbed in order to create a corresponding feature in the MultiNerve Mapper. Our goal here is to formalize this intuition. For this we adapt the bottleneck distance so that it takes the staircases into account. Our results are stated for the MultiNerve Mapper, they hold the same for the Mapper with the staircase $Q^{\mathcal{I}}_{E^-}$ replaced by its extension $Q^{\mathcal{I}}_{E}$.

An extension of the bottleneck distance. Let Θ be a subset of $\mathbb{R}^2$. Given a partial matching Γ between two extended persistence diagrams $\mathrm{ExDg}, \mathrm{ExDg}'$, the Θ-*cost* of Γ is:

$$\mathrm{cost}_\Theta(\Gamma) = \max\left\{\max_{p\in\mathrm{ExDg}} \delta_{\mathrm{ExDg}}(p),\ \max_{p'\in\mathrm{ExDg}'} \delta_{\mathrm{ExDg}'}(p')\right\},$$

where:

$$\delta_{\mathrm{ExDg}}(p) = \|p-p'\|_\infty \text{ if } \exists p' \in \mathrm{ExDg}' \text{ such that } (p,p')\in\Gamma \text{ and } d_\infty(p,\Theta) \text{ otherwise,}$$

$$\delta_{\mathrm{ExDg}'}(p') = \|p-p'\|_\infty \text{ if } \exists p \in \mathrm{ExDg} \text{ such that } (p,p')\in\Gamma \text{ and } d_\infty(p',\Theta) \text{ otherwise.}$$

The bottleneck distance becomes:

$$d_{\mathrm{b},\Theta}(\mathrm{ExDg},\mathrm{ExDg}') = \inf_\Gamma\ \mathrm{cost}_\Theta(\Gamma),$$

where Γ ranges over all partial matchings between ExDg and ExDg$'$. This is again a pseudometric and not a metric. Note that the usual bottleneck distance is obtained by taking Θ to be the diagonal Δ. Given a gomic $\mathcal{I}$, we choose different sets Θ depending on the types of the points in the two diagrams. More precisely, we define the distance between signatures as follows:

Definition 4.4.1. *Given a gomic $\mathcal{I}$, we define the distance $d_\mathcal{I}$ between extended persistence diagrams* ExDg, ExDg$'$ *as:*

$$d_\mathcal{I}(\mathrm{ExDg},\mathrm{ExDg}') = \max\left\{d_{\mathrm{b},\mathrm{Q}^\mathcal{I}_\mathrm{O}}(\mathrm{Ord},\mathrm{Ord}'),\ d_{\mathrm{b},\mathrm{Q}^\mathcal{I}_{\mathrm{E}^-}}(\mathrm{Ext},\mathrm{Ext}'),\ d_{\mathrm{b},\mathrm{Q}^\mathcal{I}_\mathrm{R}}(\mathrm{Rel},\mathrm{Rel}')\right\}. \tag{4.4}$$

The distance $d_\mathcal{I}$ *stabilizes* the (MultiNerve) Mappers, as stated in the following theorem:

Theorem 4.4.2. *Given a topological space X, Morse-type functions $f,g : X \to \mathbb{R}$ and a gomic $\mathcal{I}$ of granularity at most $\epsilon > 0$, the following stability inequality holds:*

$$d_\mathcal{I}(\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I})),\mathrm{ExDg}(\mathrm{M}_g(X,\mathcal{I}))) \le d_\mathcal{I}(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\mathrm{ExDg}(\overline{\mathrm{M}}_g(X,\mathcal{I}))) \le \|f-g\|_\infty. \tag{4.5}$$

Moreover, $d_\mathcal{I}$ and d_b are related as follows:

$$d_\mathrm{b}(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\mathrm{ExDg}(\overline{\mathrm{M}}_g(X,\mathcal{I}))) \le \frac{\varepsilon}{2} + d_\mathcal{I}(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\mathrm{ExDg}(\overline{\mathrm{M}}_g(X,\mathcal{I}))). \tag{4.6}$$

$$d_\mathrm{b}(\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I})),\mathrm{ExDg}(\mathrm{M}_g(X,\mathcal{I}))) \le \varepsilon + d_\mathcal{I}(\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I})),\mathrm{ExDg}(\mathrm{M}_g(X,\mathcal{I}))). \tag{4.7}$$

Note that Theorem 4.4.2 can be readily extended to Morse-type functions with different domains using results in [37]. In that case, the upper bound depends on the Gromov-Hausdorff distance between the domains.

The proof of Theorem 4.4.2 relies on the following monotonicity property, which is immediate:

[2] $Q^\mathcal{I}_O$, $Q^\mathcal{I}_{E^-}$ or $Q^\mathcal{I}_R$, depending on the type of the point.

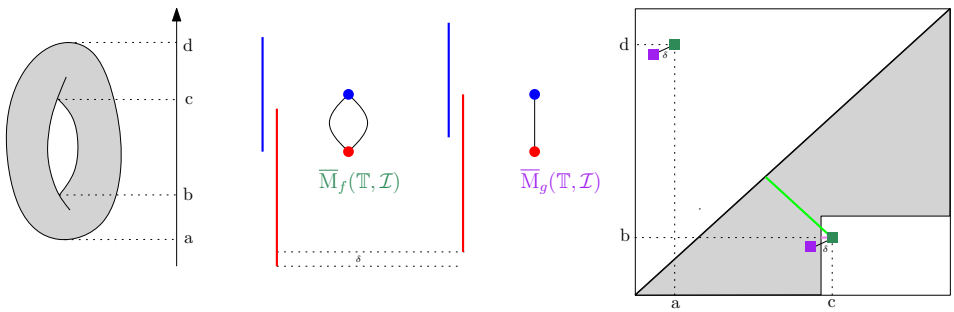

Figure 4.7: We compute the MultiNerve Mapper of the height function f on the torus $\mathbb{T}$, given a gomic $\mathcal{I}$ with two intervals. We also compute the MultiNerve Mapper of a perturbed function g such that $\|f-g\|_\infty \leq \delta$. We plot the extended persistence diagrams of $\tilde{f}$ (dark green) and $\tilde{g}$ (purple). Note that the signature of $\overline{\mathrm{M}}_g(\mathbb{T},\mathcal{I})$ is obtained by removing the purple point beneath the diagonal since it belongs to a staircase, while the signature of $\overline{\mathrm{M}}_f(\mathbb{T},\mathcal{I})$ is equal to $\mathrm{ExDg}(\tilde{f})$. If we used the bottleneck distance to compare the two signatures, their distance would be equal to the distance to the diagonal of the dark green point beneath Δ (green segment), which can be arbitrarily large, while, using $d_{\mathcal{I}}$, their distance becomes the distance of the same point to the staircase (tiny pink segment), which is bounded by δ.

Lemma 4.4.3. *Let $\Theta \subseteq \mathbb{R}^2$ be in the closure of $\Theta' \subseteq \mathbb{R}^2$. Then,*

$$d_{\Theta'}(\mathrm{ExDg},\mathrm{ExDg}') \leq d_{\Theta}(\mathrm{ExDg},\mathrm{ExDg}') \leq d_{\Theta'}(\mathrm{ExDg},\mathrm{ExDg}') + d_{\mathrm{H}}(\Theta,\Theta'),$$

where d_{H} denotes the Hausdorff distance in the ℓ_∞-norm.

Proof of Theorem 4.4.2. Equation (4.6) and (4.7) are direct applications of Lemma 4.4.3. Equation (4.5) is proven by the following sequence of (in)equalities:

$$\begin{aligned} d_{\mathcal{I}}(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\mathrm{ExDg}(\overline{\mathrm{M}}_g(X,\mathcal{I}))) &= d_{\mathcal{I}}(\mathrm{ExDg}(\tilde{f}),\mathrm{ExDg}(\tilde{g})) \\ &\leq d_{\mathrm{b},\Delta}(\mathrm{ExDg}(\tilde{f}),\mathrm{ExDg}(\tilde{g})) = d_{\mathrm{b}}(\mathrm{ExDg}(\tilde{f}),\mathrm{ExDg}(\tilde{g})) \\ &\leq d_{\mathrm{b}}(\mathrm{ExDg}(f),\mathrm{ExDg}(g)) \\ &\leq \|f-g\|_\infty. \end{aligned}$$

The first equality comes from the observation that the points of $\mathrm{ExDg}(\tilde{f}) \sqcup \mathrm{ExDg}(\tilde{g})$ that lie inside their corresponding staircase can be left unmatched and have a zero cost in the matching, so removing them as in (4.2) does not change the bottleneck cost. The first inequality follows from Lemma 4.4.3 since the diagonal Δ is included in the closure of each of the staircases. The second inequality follows from Theorem 2.4.4 and the fact that the matchings only match points of the same type (ordinary, extended, relative) and of the same homological dimension. The last inequality comes from Theorem 2.3.1. □

Interpretation of the stability. Note that the bottleneck distance d_{b} is unstable in this context—see Figure 4.7. The theorem allows us to make some interesting claims. For instance, denoting by $Q_p^{\mathcal{I}}$ the staircase corresponding to the type of a diagram point p, the quantity

$$d_{\mathcal{I}}(\mathrm{ExDg},\emptyset) = \max_{p\in\mathrm{ExDg}} d_\infty(p,Q_p^{\mathcal{I}})$$

measures the amount by which the diagram ExDg must be perturbed in the metric $d_{\mathcal{I}}$ in order to bring all its points to the staircase. Hence, by Theorem 4.4.2, given a pair (X,f), the quantity

$$d_{\mathcal{I}}(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\emptyset) = \max_{p\in\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I}))} d_\infty(p,Q_p^{\mathcal{I}})$$

is a lower bound on the amount by which f must be perturbed in the supremum norm in order to remove all the features (branches and holes) from the MultiNerve Mapper. Conversely,

$$\min_{p\in\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I}))} d_\infty(p,Q_p^{\mathcal{I}})$$

is a lower bound on the maximum amount of perturbation allowed for f if one wants to preserve all the features in the MultiNerve Mapper no matter what. Note that this does not prevent other features from appearing. The quantity that controls those is related to the points of $\mathrm{ExDg}(\tilde{f})$ (including diagonal points) that lie in the staircases. More precisely, the quantity

$$\min_{p \in \mathrm{ExDg}(\tilde{f}) \cup \Delta} d_\infty(p, \partial Q_p^{\mathcal{I}} \setminus \Delta)$$

is a lower bound on the maximum amount by which f can be perturbed if one wants to preserve the structure (set of features) of the MultiNerve Mapper no matter what. Note that this lower bound is in fact zero since $\partial Q_O^{\mathcal{I}} \setminus \Delta$ and $\partial Q_R^{\mathcal{I}} \setminus \Delta$ come arbitrarily close to the diagonal Δ (recall Figure 4.3). This means that, as small as the perturbation of f may be, it can always make new branches appear in the MultiNerve Mapper. However, it will not impact the set of holes if its amplitude is less than

$$\min_{p \in \mathrm{Ext}(\tilde{f}) \cup \Delta} d_\infty(p, \partial Q_{E^-}^{\mathcal{I}} \setminus \Delta).$$

From this discussion we derive the following rule of thumb: having small overlaps between the intervals of the gomic helps capture more features (branches and holes) of the Reeb graph in the (MultiNerve) Mapper; conversely, having large overlaps helps prevent new holes from appearing in the (MultiNerve) Mapper under small perturbations of the function. This is an important trade-off to consider in applications.

4.5 Stability with respect to perturbations of the cover

Let us now fix the pair (X, f) and consider varying gomics. For each choice of gomic, Eqs. (4.2)-(4.3) tell which points of the diagram $\mathrm{ExDg}(f)$ end up in the diagram of the (MultiNerve) Mapper and thus participate in its structure. We aim for a quantification of the extent to which this structure may change as the gomic is perturbed. For this we adopt the dual point of view: for any two choices of gomics, we want to use the points of the diagram $\mathrm{ExDg}(f)$ to assess the degree by which the gomics differ. This is a reversed situation compared to Section 4.4, where the gomic was fixed and was used to assess the degree by which the persistence diagrams of two functions differed.

A distance between gomics. The diagram points that discriminate between the two gomics are the ones located in the symmetric difference of the staircases, since they witness that the symmetric difference is non-empty. Moreover, their ℓ^∞-distances to the staircase of the other gomic provide a lower bound on the Hausdorff distance between the two staircases and thus quantify the extent to which the two covers differ. We formalize this intuition as follows: given a persistence diagram ExDg and two gomics $\mathcal{I}, \mathcal{J}$, we consider the quantity:

$$d_{\mathrm{ExDg}}(\mathcal{I}, \mathcal{J}) = \max_{*\in\{O,E^-,R\}} \left\{ \sup_{p \in \mathrm{ExDg}^* \cap (Q_*^{\mathcal{I}} \triangle Q_*^{\mathcal{J}})} \max\left\{ d_\infty(p, Q_*^{\mathcal{I}}),\, d_\infty(p, Q_*^{\mathcal{J}}) \right\} \right\}, \tag{4.8}$$

where $\triangle$ denotes the symmetric difference, where ExDg^* stands for the subdiagram of ExDg of the right type (Ord, Ext or Rel), and where we adopt the convention that $\sup_{p\in\emptyset} \ldots$ is zero instead of infinite. Note that there is always one of the two terms in (4.8) that is zero since the supremum is taken over all points that lie in the symmetric difference of the staircases. Deriving an upper bound on $d_{\mathrm{ExDg}}(\mathcal{I}, \mathcal{J})$ in terms of the Hausdorff distances between the staircases is straightforward, since the supremum in (4.8) is taken over points that lie in the symmetric difference between the staircases:

$$d_{\mathrm{ExDg}}(\mathcal{I}, \mathcal{J}) \leq \max_{*\in\{O,E^-,R\}} d_{\mathrm{H}}(Q_*^{\mathcal{I}},\, Q_*^{\mathcal{J}}),$$

where d_{H} stands for the Hausdorff distance in the ℓ^∞-norm. The connection to the MultiNerve Mapper appears when we take ExDg to be the persistence diagram of the induced map $\tilde{f}$ defined on the Reeb graph $\mathrm{R}_f(X)$. Indeed, we have

$$\mathrm{Ord}(\tilde{f}) \cap (Q_O^{\mathcal{I}} \triangle Q_O^{\mathcal{J}}) = (\mathrm{Ord}(\tilde{f}) \cap Q_O^{\mathcal{I}}) \triangle (\mathrm{Ord}(\tilde{f}) \cap Q_O^{\mathcal{J}}) = \mathrm{Ord}(\overline{\mathrm{M}}_f(X, \mathcal{I})) \triangle \mathrm{Ord}(\overline{\mathrm{M}}_f(X, \mathcal{J})),$$

where the second equality follows from the definition of the signature of the MultiNerve Mapper given in (4.2). Similar equalities can be derived with Ext and Rel. Thus, $d_{\mathrm{ExDg}(\tilde{f})}(\mathcal{I},\mathcal{J})$ quantifies the proximity of each signature to the other staircase. In particular, having $d_{\mathrm{ExDg}(\tilde{f})}(\mathcal{I},\mathcal{J}) = 0$ means that there are no diagram points in the symmetric difference, so the two gomics are equivalent from the viewpoint of the structure of the MultiNerve Mapper. Differently, having $d_{\mathrm{ExDg}(\tilde{f})}(\mathcal{I},\mathcal{J}) > 0$ means that the structures of the two MultiNerve Mappers differ, and the value of $d_{\mathrm{ExDg}(\tilde{f})}(\mathcal{I},\mathcal{J})$ quantifies by how much the covers should be perturbed to make the two multigraphs isomorphic. Furthermore, we have the following upper bound on this quantity:

Theorem 4.5.1. *Given a Morse-type function $f : X \to \mathbb{R}$, for any gomics $\mathcal{I},\mathcal{J}$,*

$$d_{\mathrm{ExDg}(\tilde{f})}(\mathcal{I},\mathcal{J}) \leq \max_{*\in\{O,E^-,R\}} d_{\mathrm{H}}(Q^{\mathcal{I}}_*,\, Q^{\mathcal{J}}_*),$$

where $\tilde{f}$ is the induced map defined on the Reeb graph $\mathrm{R}_f(X)$.

Tightness. It is easy to build examples where the upper bound is tight, for instance by placing a diagram point at a corner of one of the staircases[3]. On the other hand, there are obvious cases where the bound is not tight, for instance we have $d_{\mathrm{ExDg}(\tilde{f})}(\mathcal{I},\mathcal{J}) = 0$ as soon as there are no diagram points in the symmetric difference, whereas the symmetric difference itself may not be empty. What the upper bound measures depends on the subdiagram. For instance, for $* = E^-$, we defined $Q^{\mathcal{I}}_{E^-}$ to be the set $\bigcup_{(a,b)\in\mathcal{I}}\{(x,y) \in \mathbb{R}^2 \,:\, a \leq y < x \leq b\}$, so $d_{\mathrm{H}}(Q^{\mathcal{I}}_{E^-},\, Q^{\mathcal{J}}_{E^-})$ measures the supremum of the differences between the intervals in one cover to their closest interval in the other cover:

$$d_{\mathrm{H}}(Q^{\mathcal{I}}_{E^-},\, Q^{\mathcal{J}}_{E^-}) = \max\left\{\sup_{(a,b)\in\mathcal{I}}\inf_{(c,d)\in\mathcal{J}}\max\{|a-c|,\,|b-d|\},\ \sup_{(c,d)\in\mathcal{J}}\inf_{(a,b)\in\mathcal{I}}\max\{|a-c|,\,|b-d|\}\right\}.$$

Similar formulas can be derived for the other subdiagrams.

4.6 Convergence in the functional distortion distance

Since d_{b} is merely a pseudometric, the relationship between the (MultiNerve) Mapper and the Reeb graph is only partially explained by Theorem 4.3.3. In this section, we bound the *functional distortion distance* d_{FD} between the (MultiNerve) Mapper and the Reeb graph, and we provide an alternative proof of Theorem 4.3.3 as a byproduct. To this end, we connect the (MultiNerve) Mapper and the Reeb graph through the operators of Section 3.1, with which we can control the functional distortion distance.

4.6.1 Operators on MultiNerve Mapper

We first provide invariance results for MultiNerve Mappers computed on telescopes as defined in Section 3.1. The result is stated in a way that is adapted to its use in the following sections. The conclusion would still hold under somewhat weaker assumptions.

Proposition 4.6.1. *Let T be a telescope, π_2 be the projection onto the second coordinate, and $\mathcal{I}$ be a gomic of $\mathrm{im}(\pi_2)$. Let $\mathrm{End}(\mathcal{I})$ denote the set of endpoints of intervals of $\mathcal{I}$, sorted in ascending order. All isomorphisms mentioned in the following items are in the category of combinatorial multigraphs.*

(i) *Let $a \leq b$ such that there exists an interval $I \in \mathcal{I}$ for which a, b belong to either $I^-_\cap$, $\tilde{I}$ or $I^+_\cap$. Then, $\overline{\mathrm{M}}_{\pi_2}(\mathrm{Merge}_{a,b}(T),\mathcal{I})$ is isomorphic to $\overline{\mathrm{M}}_{\pi_2}(T,\mathcal{I})$.*

(ii) *Let $a_i \in \mathrm{Crit}(T) \setminus \mathrm{End}(\mathcal{I})$, and $a < a_i < b$ with a, b consecutive in $\mathrm{End}(\mathcal{I})$. If $a_{i-1} < a < b < a_{i+1}$ and $0 < \varepsilon < \min\{a_i - a, b - a_i\}$, then $\overline{\mathrm{M}}_{\pi_2}(\mathrm{Split}_{\varepsilon,a_i}(T),\mathcal{I})$ is isomorphic to $\overline{\mathrm{M}}_{\pi_2}(T,\mathcal{I})$.*

(iii) *Let $a_i \in \mathrm{Crit}(T) \setminus \mathrm{End}(\mathcal{I})$, and $b < a_i < c < d$ with b, c, d consecutive in $\mathrm{End}(\mathcal{I})$. If a_i is an up-fork, $(b, c) = I \cap J$ is an intersection, and $c - a_i < \varepsilon < \min\{d, a_{i+1}\} - a_i$, then $\overline{\mathrm{M}}_{\pi_2}(\mathrm{Shift}_{\varepsilon,a_i}(T),\mathcal{I})$ is isomorphic to $\overline{\mathrm{M}}_{\pi_2}(T,\mathcal{I})$.*

[3] Which is easily done by choosing suitable critical values as coordinates for this point.

(iv) *Let* $a_i \in \mathrm{Crit}(T) \setminus \mathrm{End}(\mathcal{I})$*, and* $a < b < a_i < c$ *with* a, b, c *consecutive in* $\mathrm{End}(\mathcal{I})$*. If* a_i *is a down-fork,* $(b, c) = I \cap J$ *is an intersection, and* $\max\{a, a_{i-1}\} - a_i < \varepsilon < b - a_i$*, then* $\overline{\mathrm{M}}_{\pi_2}(\mathrm{Shift}_{\varepsilon,a_i}(T), \mathcal{I})$ *is isomorphic to* $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$*.*

Proof. Under the assumptions given by each item, the connected components in every intersection $I \cap J$, $I, J \in \mathcal{I}$ and in every element $I \in \mathcal{I}$ remain the same after each operation. Given any intersection $K = I \cap J$, $I, J \in \mathcal{I}$, or interval $K = I \in \mathcal{I}$, we recall that T^K denotes $\pi_1 \circ \pi_2^{-1}(K)$. Then, we have:

(i) - (ii) T^K deform retracts onto $(\mathrm{Merge}_{a,b}(T))^K$ and $(\mathrm{Split}_{\varepsilon,a_i}(T))^K$ deform retracts onto T^K;

(iii) - (iv) The Shifts move the up-fork to the upper proper subinterval, and the down-fork to the lower proper subinterval, which preserves the connected components in each of the two intervals as well as in their intersection.

Thus, the MultiNerve Mapper is not changed by any of the aforementioned operations. □

4.6.2 Connection between the (MultiNerve) Mapper and the Reeb graph.

In this section, we describe a sequence of metric spaces linking the MultiNerve Mapper and the Reeb graph. Let $f : X \to \mathbb{R}$ be of Morse type, and let $\mathcal{I}$ be a gomic of $\mathrm{im}(f)$. Let $T(X, f)$ be the corresponding telescope. The idea is to move all critical values out of the intersection preimages $f^{-1}(I \cap J)$, so that the MultiNerve Mapper and the Reeb graph become isomorphic. For any interval $I \in \mathcal{I}$, we let $a_{\tilde{I}} < b_{\tilde{I}}$ be the endpoints of its proper subinterval $\tilde{I}$, so we have $\tilde{I} = [a_{\tilde{I}}, b_{\tilde{I}}]$. For any non-empty intersection $I \cap J$, we fix a subinterval $[a_{I\cap J}, b_{I\cap J}] \subset I \cap J$ such that every critical value within $I \cap J$ falls into $[a_{I\cap J}, b_{I\cap J}]$ (which is possible because f is of Morse type hence has finitely many critical values). We then define three different operations individually as follows:

- $\mathrm{Merge}_{\mathcal{I}}$ is the composition of all the $\mathrm{Merge}_{a_{\tilde{I}},b_{\tilde{I}}}$, $I \in \mathcal{I}$, and of all the $\mathrm{Merge}_{a_{I\cap J},b_{I\cap J}}$, $I, J \in \mathcal{I}$ and $I \cap J \neq \emptyset$. All these functions commute, so their composition is well-defined. The same holds for the following compositions.
- $\mathrm{Split}_{\mathcal{I}}$ is the composition of all the $\mathrm{Split}_{\varepsilon,\bar{a}}$ with $\bar{a}$ a critical value after $\mathrm{Merge}_{\mathcal{I}}$ (therefore not an interval endpoint) and $\varepsilon > 0$ such that the assumptions of Proposition 4.6.1 (ii) are satisfied.
- $\mathrm{Shift}_{\mathcal{I}}$ is the composition of all the $\mathrm{Shift}_{\varepsilon,\bar{a}_+}$ with $\bar{a}_+$ an up-fork critical value after the $\mathrm{Split}_{\mathcal{I}}$ and $\varepsilon > 0$ such that the assumptions of Proposition 4.6.1 (iii) are satisfied, and of all the $\mathrm{Shift}_{\varepsilon,\bar{a}_-}$ with $\bar{a}_-$ a down-fork critical value after the $\mathrm{Split}_{\mathcal{I}}$ and $\varepsilon < 0$ such that the assumptions of Proposition 4.6.1 (iv) are satisfied. After $\mathrm{Shift}_{\mathcal{I}}$ there are no more critical values located in the intersections of consecutive intervals of $\mathcal{I}$.
- $\mathrm{Merge}'_{\mathcal{I}}$ is the composition of all the $\mathrm{Merge}_{a_{\tilde{I}},b_{\tilde{I}}}$, $I \in \mathcal{I}$.

We can now define our sequence of intermediate spaces:

Definition 4.6.2. *Let* X *be a topological space,* $f : X \to \mathbb{R}$ *be a Morse-type function, and* $\mathcal{I}$ *be a gomic of* $\mathrm{im}(f)$*. Let* $T(X, f)$ *be the telescope associated to* f*. We define the telescope* $\bar{T}_{\mathcal{I}}$ *with:*

$$\bar{T}_{\mathcal{I}}(X, f) = \mathrm{Merge}'_{\mathcal{I}} \circ \mathrm{Shift}_{\mathcal{I}} \circ \mathrm{Split}_{\mathcal{I}} \circ \mathrm{Merge}_{\mathcal{I}}(T(X, f)).$$

We also let $\bar{f}_{\mathcal{I}}$ *denote the projection of* $\bar{T}_{\mathcal{I}}$ *onto the second factor.*

See Figure 4.8 for an illustration of this sequence of transformations. When often write $\bar{T}_{\mathcal{I}}$ instead of $\bar{T}_{\mathcal{I}}(X, f)$ when the pair (X, f) is clear from the context. In the following, we identify the pair (T, π_2) with (X, f) since they are isomorphic in the category of $\mathbb{R}$-constructible spaces. We also let $\bar{\bar{f}}_{\mathcal{I}} : \mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}) \to \mathbb{R}$ denote the induced map defined on the Reeb graph of $\bar{T}_{\mathcal{I}}$.

Thanks to Proposition 4.6.1 and the choice of the $a_{\tilde{I}}, b_{\tilde{I}}, a_{I\cap J}, b_{I\cap J}, \epsilon$ in the definitions of $\mathrm{Merge}_{\mathcal{I}}$, $\mathrm{Split}_{\mathcal{I}}, \mathrm{Shift}_{\mathcal{I}}$ and $\mathrm{Merge}'_{\mathcal{I}}$, we provide Lemma 4.6.3 below, which states that the MultiNerve Mapper is not affected by this sequence of transformations.

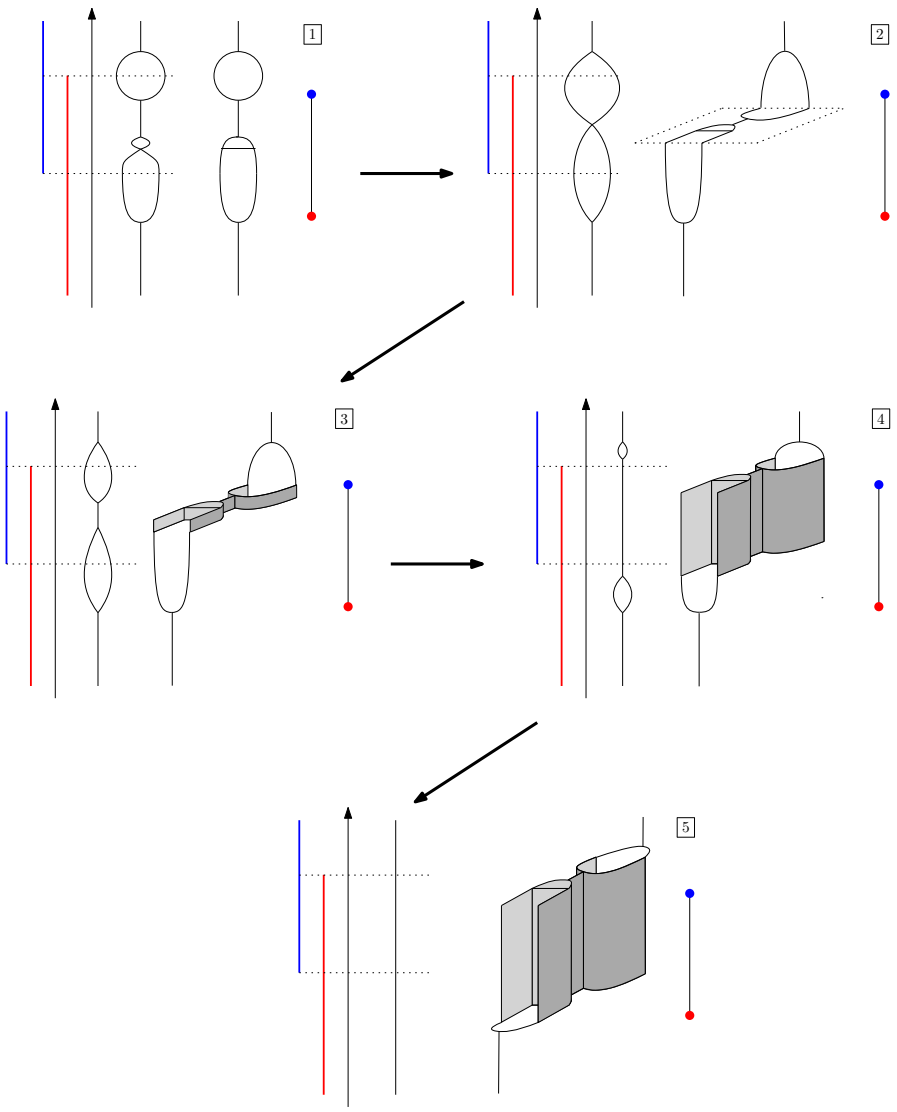

Figure 4.8: Illustration of the sequence of transformations in (4.6.2) on the features located in an interval intersection. For each figure, we display the original space (middle), its Reeb graph (left) and its MultiNerve Mapper (right).

Lemma 4.6.3. *For $(\bar{T}_{\mathcal{I}}, \bar{f}_{\mathcal{I}})$ defined as in Definition 4.6.2, $\overline{\mathrm{M}}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}, \mathcal{I})$ and $\overline{\mathrm{M}}_f(X, \mathcal{I})$ are isomorphic as combinatorial multigraphs.*

This allows us to prove the following result, which states that the MultiNerve Mapper $\overline{\mathrm{M}}_f(X, \mathcal{I})$ is actually the same object than the perturbed Reeb graph $\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}})$.

Theorem 4.6.4. *For $(\bar{T}_{\mathcal{I}}, \bar{f}_{\mathcal{I}})$ defined as in Definition 4.6.2, $\overline{\mathrm{M}}_f(X, \mathcal{I})$ and $\mathcal{C}\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}})$ are isomorphic as combinatorial multigraphs.*

We know from Lemma 4.6.3 that $\overline{\mathrm{M}}_f(X, \mathcal{I})$ and $\overline{\mathrm{M}}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}, \mathcal{I})$ are isomorphic as combinatorial multigraphs. Theorem 4.6.4 is then a consequence of the following result, whose hypothesis is satisfied by the $\bar{T}_{\mathcal{I}}$ of Definition 4.6.2:

Lemma 4.6.5. *Let T be a telescope and let $\pi_2 : T \to \mathbb{R}$ be the projection onto the second factor. Suppose that every proper subinterval $\tilde{I}$ in the cover $\mathcal{I}$ contains exactly one critical value of π_2, and that*

the intersections $I \cap J$ contain none. Then, $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ and $\mathcal{C}\mathrm{R}_{\pi_2}(T)$ are isomorphic as combinatorial multigraphs.

Proof. The nodes of $\mathcal{C}\mathrm{R}_{\pi_2}(T)$ represent the connected components of the preimages of all critical values of π_2, while the nodes of $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ represent the connected components of the preimages of all $I \in \mathcal{I}$. The hypothesis of the lemma implies that there is exactly one critical value per interval $I \in \mathcal{I}$, hence the nodes of $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ and of $\mathcal{C}\mathrm{R}_{\pi_2}(T)$ are in bijection. Meanwhile, the edges of $\mathcal{C}\mathrm{R}_{\pi_2}(T)$ are given by the connected components of the $Y_i \times [a_i, a_{i+1}]$. Since the proper subintervals contain one critical value each and the $I \cap J$ contain none, the pullbacks of all intersections of consecutive intervals also span the $Y_i \times [a_i, a_{i+1}]$. Hence, the edges of $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ are in bijection with the ones of $\mathcal{C}\mathrm{R}_{\pi_2}(T)$. Moreover, their endpoints are defined in both cases by the ϕ_i and ψ_i. Hence the multigraph isomorphism. □

In passing, it is interesting to study the behavior of the MultiNerve Mapper as the hypothesis of the lemma is weakened. For instance:

Lemma 4.6.6. *Let T be a telescope and let $\pi_2 : T \to \mathbb{R}$ be the projection onto the second factor. Suppose that every interval I in the cover $\mathcal{I}$ contains at most one critical value of π_2. Then, $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ is obtained from $\mathcal{C}\mathrm{R}_{\pi_2}(T)$ by splitting some vertices into two and by subdividing some edges once.*

Thus, the MultiNerve Mapper may non longer be 'exactly' isomorphic to the combinatorial Reeb graph (counter-examples are easy to build, by making some of the critical values fall into intersections of intervals in the cover), however it is still isomorphic to it up to vertex splits and edge subdivisions, which are topologically trivial modifications.

Proof of Lemma 4.6.6. The proof is constructive and it proceeds in 3 steps:
1. For every interval $I \in \mathcal{I}$ that does not contain a critical value, add a dummy critical value (with identities as connecting maps) in the proper subinterval $\tilde{I}$. The effect on the Mapper is null, while the effect on the Reeb graph is to subdivide once each edge crossing the dummy critical value. At this stage, every interval of $\mathcal{I}$ contains exactly one critical value. For simplicity we identify T with the new telescope.
2. For every interval $I \in \mathcal{I}$ whose corresponding critical value does not lie in the proper subinterval $\tilde{I}$ but rather in some intersection $I \cap J$ (defined uniquely since $\mathcal{I}$ is a gomic), merge I and J into a single interval $I \cup J$. The coarser cover $\mathcal{J}$ thus obtained is still a gomic and it has the extra property that every proper subinterval contains exactly one critical value and every intersection contains none. Then, by Lemma 4.6.5, the MultiNerve Mapper $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{J})$ is isomorphic to the combinatorial Reeb graph $\mathcal{C}\mathrm{R}_{\pi_2}(T)$.
3. There remains to study the differences between $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ and $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{J})$. The only difference between the two covers is that some isolated pairs of intervals (I, J) have been merged because their intersection $I \cap J$ contained a critical value a_i. For every such pair, there are as many connected components in the preimage $\pi_2^{-1}(I)$ as in $\pi_2^{-1}(J)$ as in $\pi_2^{-1}(I \cap J)$ as in $\pi_2^{-1}(I \cup J)$ because $I \cup J$ contains no critical value other than a_i. Hence, every vertex of $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{J})$ corresponding to a connected component of $\pi_2^{-1}(I \cup J)$ is split into two in $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$. Moreover, the two copies are connected by a single edge, given by the corresponding connected component of $\pi_2^{-1}(I \cap J)$. Now, assuming without loss of generality that J lies above I, we have $(I \cup J)^+_\cap = J^+_\cap$, which by assumption contains no critical value, so the connections between the vertex copy corresponding to $\pi_2^{-1}(J)$ and the vertices lying above it in $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ are the same as the connections between the original vertex and the vertices lying above it in $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{J})$. Similarly, $(I \cup J)^-_\cap = I^-_\cap$ contains no critical value by assumption, so the connections between the vertex copy corresponding to $\pi_2^{-1}(I)$ and the vertices lying below it in $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{I})$ are the same as the connections between the original vertex and the vertices lying below it in $\overline{\mathrm{M}}_{\pi_2}(T, \mathcal{J})$. □

Extension to the Mapper. Due to the simple relation between the Mapper and the MultiNerve Mapper given by Corollary 4.2.5, Theorem 4.6.4 can be extended for Mappers.

Definition 4.6.7. *Let X be a topological space, and $f : X \to \mathbb{R}$ be a Morse-type function. Let $(\bar{T}_{\mathcal{I}}, \bar{f}_{\mathcal{I}})$ be defined as in Definition 4.6.2. Let $\mathrm{Cyl}(\bar{T}_{\mathcal{I}})$ be the set of the connected components of the cylinders of*

$\bar{T}_{\mathcal{I}}$. *We define the equivalence relation* $\sim$ *between elements of* $\mathrm{Cyl}(\bar{T}_{\mathcal{I}})$ *as:*

$$C \sim C' \Leftrightarrow \begin{cases} C, C' \text{ are connected components of the same cylinder} \\ \phi_i(C \times \{a_i\}) \text{ and } \phi_i(C' \times \{a_i\}) \text{ belong to the same connected component} \\ \psi_i(C \times \{a_{i+1}\}) \text{ and } \psi_i(C' \times \{a_{i+1}\}) \text{ belong to the same connected component} \end{cases}$$

Then, we define $T_{\mathcal{I}}$ *as* $\bar{T}_{\mathcal{I}}/\sim$, *equipped with the projection onto the second factor that we call* $f_{\mathcal{I}}$.

Intuitively, we glue the pairs C, C' of connected components of the same cylinder whose images under the attaching maps are in the same connected component of the critical slice, i.e. those that induce edges with the same endpoints in the multinerve. Hence, we obtain the following corollary using Corollary 4.2.5:

Corollary 4.6.8. $\mathcal{C}\mathrm{R}_{f_{\mathcal{I}}}(T_{\mathcal{I}})$ *and* $\mathrm{M}_f(X, \mathcal{I})$ *are isomorphic as combinatorial multigraphs.*

4.6.3 Convergence results.

Recall that the d_{FD} compares metric graphs, whereas the (MultiNerve) Mappers are combinatorial graphs. However, since $\overline{\mathrm{M}}_f(X, \mathcal{I})$ and $\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}})$ are essentially the same according to Theorem 4.6.4, we can use $\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}})$ as a metric graph representation of $\overline{\mathrm{M}}_f(X, \mathcal{I})$, when computing the functional distortion distance. Note that we could also use $\mathrm{R}_{\bar{\mathsf{m}}_{\mathcal{I}}}(\overline{\mathrm{M}}_f(X, \mathcal{I}))$ since it is isomorphic to $\overline{\mathrm{M}}_f(X, \mathcal{I})$ as well according to Lemma 4.3.2, but its connection to $\mathrm{R}_f(X)$ is unclear. On the opposite, even though d_{FD} is most of the time untractable, its computation is possible with $\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}})$ thanks to the sequence of transformations of Definition 4.6.2. We will see at the end of the section that $\bar{\mathsf{m}}_{\mathcal{I}}$ and $\bar{\tilde{f}}_{\mathcal{I}}$ actually coincide on $\overline{\mathrm{M}}_f(X, \mathcal{I})$.

Theorem 4.6.10 below shows that $\mathrm{R}_{\bar{\mathsf{m}}_{\mathcal{I}}}(\overline{\mathrm{M}}_f(X, \mathcal{I}))$ is close to $\mathrm{R}_f(X)$ if $\mathcal{I}$ has a small granularity. To prove it, we use the following lemma, whose proof is just a simple extension of the one of Proposition 3.2.3:

Lemma 4.6.9. *Let* S *be a set of pairwise disjoint bounded open intervals, and let* Merge_S *be defined as the composition of all* $\mathrm{Merge}_{a,b}$, $(a, b) \in S$. *Let* R_g *be a Reeb graph such that* $\mathrm{Crit}(\tilde{g}) \subset \bigcup_{I \in S} I$ *and let* $\mathrm{R}_{g'}$ *be the Reeb graph of the telescope* $\mathrm{Merge}_S(\mathrm{R}_g)$. *Then* $d_{\mathrm{FD}}(\mathrm{R}_g, \mathrm{R}_{g'}) \leq \sup\{\mathrm{length}(I) : I \in S\}$.

Given a gomic $\mathcal{I}$, we let $\epsilon_1(\mathcal{I}) = \sup\{\mathrm{length}(\tilde{I}) : I \in \mathcal{I}\}$ and $\epsilon_2(\mathcal{I}) = \sup\{\mathrm{length}(I \cap J) : I, J \in \mathcal{I}\}$. Note that $\epsilon_1(\mathcal{I})$ and $\epsilon_2(\mathcal{I})$ can be thought of as different types of granularity measures of $\mathcal{I}$. They are both bounded from above by the granularity of $\mathcal{I}$ as defined in Section 2.5.

Theorem 4.6.10. *Suppose the granularity of the gomic* $\mathcal{I}$ *is at most* ε. *For* $(\bar{T}_{\mathcal{I}}, \bar{f}_{\mathcal{I}})$ *defined as in Definition 4.6.2, we have* $d_{\mathrm{FD}}(\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}), \mathrm{R}_f(X)) \leq \epsilon_1(\mathcal{I}) + \epsilon_2(\mathcal{I}) + \max\{\epsilon_1(\mathcal{I}), \epsilon_2(\mathcal{I})\} \leq 3\epsilon$.

Moreover, for $(T_{\mathcal{I}}, f_{\mathcal{I}})$ *defined as in Definition 4.6.7, we have* $d_{\mathrm{FD}}(\mathrm{R}_{f_{\mathcal{I}}}(T_{\mathcal{I}}), \mathrm{R}_f(X)) \leq 7\varepsilon/2$.

Proof. We start with the MultiNerve Mapper. By the triangle inequality, it suffices to bound the functional distortion distance for each of the four operations $\mathrm{Merge}_{\mathcal{I}}$, $\mathrm{Shift}_{\mathcal{I}}$, $\mathrm{Split}_{\mathcal{I}}$ and $\mathrm{Merge}'_{\mathcal{I}}$ individually. Let R_1 be the Reeb graph of the telescope $\mathrm{Merge}_{\mathcal{I}}(\mathrm{R}_f(X))$. Similarly, let R_2 be the Reeb graph of $\mathrm{Split}_{\mathcal{I}}(\mathrm{R}_1)$, R_3 be the Reeb graph of $\mathrm{Shift}_{\mathcal{I}}(\mathrm{R}_2)$ and $\mathrm{R}_4 = \mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}})$ be the Reeb graph of $\mathrm{Merge}'_{\mathcal{I}}(\mathrm{R}_3)$. Examples of such Reeb graphs can be seen in the left parts of Figure 4.8.

Then we have $d_{\mathrm{FD}}(\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}), \mathrm{R}_f(X)) \leq d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_1) + d_{\mathrm{FD}}(\mathrm{R}_1, \mathrm{R}_2) + d_{\mathrm{FD}}(\mathrm{R}_2, \mathrm{R}_3) + d_{\mathrm{FD}}(\mathrm{R}_3, \mathrm{R}_4)$.

- By Lemma 4.6.9, we have $d_{\mathrm{FD}}(\mathrm{R}_f(X), \mathrm{R}_1) \leq \max\{\epsilon_1(\mathcal{I}), \epsilon_2(\mathcal{I})\}$ and $d_{\mathrm{FD}}(\mathrm{R}_3, \mathrm{R}_4) \leq \epsilon_1(\mathcal{I})$.
- Assume without loss of generality that $\mathrm{Split}_{\mathcal{I}}$ is the composition of all $\mathrm{Split}_{\alpha,\bar{a}}$, where $\bar{a}$ is a critical value of R_1. Since R_1 is obtained from R_2 by taking the composition of all $\mathrm{Merge}_{\bar{a}-\alpha,\bar{a}+\alpha}$, it follows from Lemma 4.6.9 that $d_{\mathrm{FD}}(\mathrm{R}_1, \mathrm{R}_2) \leq 2\alpha$.
- Since the assumptions of Prop. 4.6.1 (iii) and Prop. 4.6.1 (iv) are satisfied by $\mathrm{Shift}_{\mathcal{I}}$, it follows that R_2 and R_3 are isomorphic, because the number, the types and the ordering of the critical values of R_2 are preserved when transformed into R_3. It is then straightforward that the functional distortion distance between R_2 and R_3 is the maximal amplitude of the Shift operations involved. According to the assumptions of Proposition 4.6.1 (iii) and Proposition 4.6.1 (iv), these amplitudes are all bounded by $\epsilon_2(\mathcal{I})$.

The result follows by letting $\alpha \to 0$.

Concerning the Mapper, the result is obtained by adding an extra $\epsilon/2$ to the previous upper bound, which corresponds to the functional distortion distance cost of gluing edges with the same endpoints. $\square$

Note that a similar result can be obtained directly by using the convergence result of the so-called *interleaving distance*—see Theorem 4.1 in [105], and the strong equivalence between the functional distortion distance and this interleaving distance—see Theorem 14 in [10]. However, the upper bound gets larger (7ϵ) and there is no clear intuition on the Reeb graph used to represent the Mapper (also called geometric Mapper) in [105].

Finally, Theorems 4.6.10 and 2.4.10 allow us to derive the following result with the triangle inequality:

Corollary 4.6.11. *Let X be a topological space, and let $f, g : X \to \mathbb{R}$ be two Morse-type functions with continuous sections. Let $(\bar{T}_{\mathcal{I}}(X, f), \bar{f}_{\mathcal{I}})$ (resp. $(\bar{T}_{\mathcal{I}}(X, g), \bar{g}_{\mathcal{I}})$) denote the pair computed with function f (resp. g) as in Definition 4.6.2. Finally, let $\mathcal{I}$ be a gomic of granularity at most ϵ. Then:*

$$d_{\mathrm{FD}}(\mathrm{R}_{\bar{f}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}(X, f)), \mathrm{R}_{\bar{g}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}(X, g))) \leq \|f - g\|_\infty + 6\epsilon.$$

Moreover, for $(T_{\mathcal{I}}(X, f), f_{\mathcal{I}})$ and $(T_{\mathcal{I}}(X, g), g_{\mathcal{I}})$ computed as in Definition 4.6.7, we have:

$$d_{\mathrm{FD}}(\mathrm{R}_{f_{\mathcal{I}}}(T_{\mathcal{I}}(X, f)), \mathrm{R}_{g_{\mathcal{I}}}(T_{\mathcal{I}}(X, g))) \leq \|f - g\|_\infty + 7\epsilon.$$

4.6.4 An alternative proof of Theorem 4.3.3

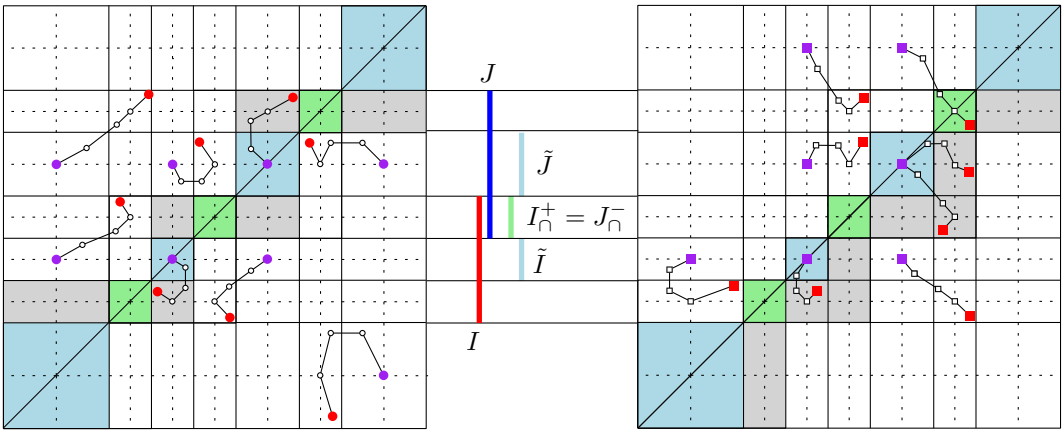

Figure 4.9: The left panel displays the trajectories of points in Ord (disks above the diagonal) and Rel (disks below the diagonal) while the right panel displays the trajectories of points in Ext. For both diagrams, the original point is red, the final point is purple, and intersections and proper subintervals are colored in green and light blue respectively.

We can prove Theorem 4.3.3 again by studying the effect of the transformation defined in Definition 4.6.2 on the extended persistence diagram of f. These effect is illustrated in Figure 4.9. There are two grids in this figure: the one with solid lines is defined by the interval endpoints, while the one with dotted lines is defined by midpoints of proper subintervals and intersections. In the following, we use the term *cell* to designate a rectangle of the first grid. Cells are closed if they correspond to proper subintervals for both coordinates, they are open if they correspond to intersections for both coordinates, and they are neither closed nor open otherwise. Blue and green cells in Figure 4.9 correspond to squares associated to a proper subinterval (blue) or intersection (green). We can now interpret the effects of the transformations in (4.6.2) on the persistence diagram visually:

- $\mathrm{Merge}_{\mathcal{I}}$ snaps all the points to the second grid by Lemma 3.1.6.

- $\mathrm{Split}_{\mathcal{I}}$ moves the points to one of the four possible quarters inside their cell, depending on the point's type by Lemma 3.1.12. More precisely, ordinary points are moved to the down-left quarter, extended points are moved to the up-left quarter, and relative points are moved to the up-right quarter.
- Then, $\mathrm{Shift}_{\mathcal{I}}$ moves the points to a neighboring cell by Lemma 3.1.14. This neighboring cell is given by the point's type (as in the case of $\mathrm{Split}_{\mathcal{I}}$) and by the coordinates of the point. For instance, an extended point (x, y) lying in the row of a green cell and in the column of another green cell, has coordinates that both belong to interval intersections. Then, this point is moved to the upper-left neighboring cell. Differently, an extended point whose abscissa (resp. ordinate) is in an intersection and whose ordinate (resp. abscissa) is not, is only moved to the left (resp. upper) cell. The same can be said for ordinary (resp. relative) points by changing up-left to down-left (resp. up-right). Points whose coordinates both belong to proper subintervals are not moved by $\mathrm{Shift}_{\mathcal{I}}$, regardless of their type.
- Finally, $\mathrm{Merge}'_{\mathcal{I}}$ re-snaps the points to the second grid by Lemma 3.1.6.

Thus, each point of $\mathrm{ExDg}(f)$ can be tracked through the successive operations of (4.6.2), and this tracking leads to the following elementary observations:

1. The points of $\mathrm{Ord}(f)$ or $\mathrm{Rel}(f)$ that end their course on the diagonal after the sequence of operations of Definition 4.6.2 disappear in $\mathrm{ExDg}(\bar{f}_{\mathcal{I}})$. This is because ordinary and relative points cannot be located on the diagonal. The rest of the points of $\mathrm{Ord}(f)$ or $\mathrm{Rel}(f)$ are preserved with the same type in $\mathrm{ExDg}(\bar{f}_{\mathcal{I}})$.
2. Differently, all the points of $\mathrm{Ext}(f)$ are preserved with the same type (Ext) in $\mathrm{ExDg}(\bar{f}_{\mathcal{I}})$. However, some of the points of $\mathrm{Ext}^-(f)$ may end their course on or across the diagonal after the sequence of operations (4.6.2), thus switching from $\mathrm{Ext}^-(f)$ to $\mathrm{Ext}^+(\bar{f}_{\mathcal{I}})$.
3. All the points lie in the rows and columns of blue cells after $\mathrm{Shift}_{\mathcal{I}}$. Therefore, the points that end their course on the diagonal after the sequence of operations of Definition 4.6.2 are the ones located in blue cells after $\mathrm{Shift}_{\mathcal{I}}$.
4. Since transfers between cells occur only during $\mathrm{Shift}_{\mathcal{I}}$, a point p that is not in a blue or green cell initially ends up in a blue cell B after $\mathrm{Shift}_{\mathcal{I}}$ if and only if:
   - p is extended and it is in the down, right, or down-right neighboring cell of B (grey cells in the right diagram of Figure 4.9), or
   - p is ordinary and it is in the up neighboring cell of B (grey cells above the diagonal in the left diagram of Figure 4.9), or
   - p is relative and it is in the down neighboring cell of B (grey cells below the diagonal in the left diagram of Figure 4.9).
5. Points that belong to a blue or green cell initially are snapped to the diagonal by $\mathrm{Merge}_{\mathcal{I}}$. Among them, those that belong to a blue cell stay there until the end, whereas those that belong to a green cell eventually leave it—they end up in a blue cell after $\mathrm{Shift}_{\mathcal{I}}$ if they are ordinary or relative, while they end up in a white cell above the diagonal if they are extended.

The outcome of these observations is the following one. Observations 1, 3, 4, 5 imply that the points of $\mathrm{Ord}(f)$ that disappear in $\mathrm{ExDg}(\bar{f}_{\mathcal{I}})$ are the ones located initially in the staircase made of the blue, green and grey areas above the diagonal in the left panel of Figure 4.9, which is nothing but $Q_O^{\mathcal{I}}$. Similarly, the points of $\mathrm{Rel}(f)$ that disappear in $\mathrm{ExDg}(\bar{f}_{\mathcal{I}})$ are the ones located initially in the staircase made of the blue, green and grey areas below the diagonal in the left panel of Figure 4.9, which is nothing but $Q_R^{\mathcal{I}}$. Finally, observations 2, 3, 4, 5 imply that the points of $\mathrm{Ext}^-(f)$ that switch to $\mathrm{Ext}^+(\bar{f}_{\mathcal{I}})$ are the ones located initially in the staircase made of the blue, green and grey areas below the diagonal in the right panel of Figure 4.9, which is nothing but $Q_{E^-}^{\mathcal{I}}$. The rest of the points of $\mathrm{ExDg}(f)$ are preserved (albeit shifted) with the same type (Ord, Rel, Ext^+, Ext^-) in $\mathrm{ExDg}(\bar{f}_{\mathcal{I}})$. Hence, there is a perfect matching between:

(i) $\mathrm{Ord}(\bar{f}_{\mathcal{I}})$ and $\mathrm{Ord}(f) \setminus Q_O^{\mathcal{I}}$ (iii) $\mathrm{Ext}^-(\bar{f}_{\mathcal{I}})$ and $\mathrm{Ext}^-(f) \setminus Q_{E^-}^{\mathcal{I}}$
(ii) $\mathrm{Rel}(\bar{f}_{\mathcal{I}})$ and $\mathrm{Rel}(f) \setminus Q_R^{\mathcal{I}}$ (iv) $\mathrm{Ext}^+(\bar{f}_{\mathcal{I}})$ and $\mathrm{Ext}^+(f) \cup (\mathrm{Ext}^-(f) \cap Q_{E^-}^{\mathcal{I}})$

This, combined with Theorem 2.4.4, is equivalent to Theorem 4.3.3. This matching also shows that the critical points of $\bar{f}_{\mathcal{I}}$ and $f_{\mathcal{I}}$ are located at the midpoints of proper subintervals of the gomic's elements. Hence, $\tilde{\bar{f}}_{\mathcal{I}}$ and $\tilde{f}_{\mathcal{I}}$ actually coincide with $\bar{\mathsf{m}}_{\mathcal{I}}$ and $\mathsf{m}_{\mathcal{I}}$, which allows us to state this final result:

Theorem 4.6.12. *Let X be a topological space and $f : X \to \mathbb{R}$ be a Morse-type function. Let $\mathcal{I}$ be a gomic with granularity at most ϵ. Let $\bar{\mathsf{m}}_{\mathcal{I}}$ and $\mathsf{m}_{\mathcal{I}}$ be as in Definition 4.3.1. Then:*

$$d_{\mathrm{b}}(\mathrm{ExDg}(\bar{\mathsf{m}}_{\mathcal{I}}), \mathrm{ExDg}(\tilde{f})) \leq \epsilon/2, \tag{4.9}$$

$$d_{\mathrm{b}}(\mathrm{ExDg}(\mathsf{m}_{\mathcal{I}}), \mathrm{ExDg}(\tilde{f})) \leq \epsilon. \tag{4.10}$$

Moreover, in both cases, the matching achieving the distance is actually a bijection preserving types.

4.7 Conclusion

In this chapter, we showed that the topological structure of the (MultiNerve) Mapper can be simply read off from the one of the Reeb graph, via an appropriate simplification of its extended persistence diagram. This simplification, namely the removal of points belonging to specific staircases, allowed us to define a natural distance between (MultiNerve) Mappers by using appropriate signatures $\mathrm{ExDg}(\overline{\mathrm{M}}_f(X, \mathcal{I}))$ and $\mathrm{ExDg}(\mathrm{M}_f(X, \mathcal{I}))$, to show that (MultiNerve) Mappers converge to their corresponding Reeb graphs (Corollary 4.3.6) and that they are stable (Theorem 4.4.2) with respect to this distance. Moreover, we also showed that (MultiNerve) Mappers actually converge to their corresponding Reeb graphs in the functional distortion distance, by using computable functions $\bar{\mathsf{m}}_{\mathcal{I}}$ and $\mathsf{m}_{\mathcal{I}}$ (Theorem 4.6.10).

Among the future perspectives of this work are the following questions:

- **Does the local equivalence hold for Mapper?** In Chapter 3, we showed that d_{b} is locally a true metric for Reeb graphs. Extending this result to Mappers would require to find an equivalent of d_{FD} that depends on the gomic $\mathcal{I}$.
- **Can our analysis be extended to multivariate function?** The main limitation of this work is to be restricted to scalar-valued functions, even though this is very common in applications. The question whether our analysis can extend to multivariate functions $f : X \to \mathbb{R}^D$, with $D > 1$, remains open, and would require to extend the space's stratifications induced by Morse-type functions. A possible way to proceed is to study the so-called *Jacobi sets* of multivariate functions [41, 68, 70, 133], and to use recent results about decomposition of multidimensional persistence modules [18, 53].

CHAPTER 5

STATISTICAL ANALYSIS AND PARAMETER SELECTION

In Chapters 3 and 4, we have seen how Reeb graphs and Mappers can be compared with adequate metrics, when they are computed on non discrete topological spaces. In this chapter, we focus on Mappers computed on discrete and finite topological spaces, i.e. point clouds. In particular, we provide approximation results (Theorems 5.1.9 and 5.1.10) controlling the distance between Mappers computed on discrete and non discrete topological spaces. Moreover, we show that *interval-* and *intersection-crossing edges* are the principal responsibles of discretization artifacts (Lemma 5.1.7 and 5.1.8). This observation allows us to study the rate of convergence of the Mapper to the Reeb graph when the cardinality of the point cloud grows to $+\infty$.

Our main result is Proposition 5.3.3, which states that the rate of convergence of the Mapper, when computed on a point cloud X_n drawn from a specific probability distribution whose support is a compact Riemannian manifold embedded in $\mathbb{R}^D$, is of the order $(\log(n)/n)^{1/d}$:

$$\mathbb{E}\left[d_{\mathrm{b}}(\mathrm{M}_f(X_n, \mathcal{I}_n), \mathrm{R}_f(X))\right] \lesssim C\omega\left(\left(\frac{\log(n)}{n}\right)^{\frac{1}{d}}\right),$$

where n is the cardinality of the point cloud, C is a constant, ω is a measure of the regularity of f (for instance $\omega(x) = cx$ when f is Lipschitz with constant c), and $\mathcal{I}_n$ is a specific cover that depends only on X_n. We show that this rate is minimax optimal, meaning that no other estimator of the Reeb graph can converge faster.

We finally use the specific cover $\mathcal{I}_n$ as a heuristic to automatically tune Mapper parameters, and we build on the convergence result to compute confidence regions for the topological features of the Mapper, hence providing statistical guarantees for all applications of Topological Data Analysis relying on Mapper, such as clustering [97, 108] and feature selection [109, 122].

Plan of the Chapter. In Section 5.1, we recall how Reeb graphs and Mappers are computed on point clouds. Then, we give an approximation inequality (Theorem 5.2.1) for the Reeb graph in Section 5.2. From this approximation result, we derive rates of convergences as well as candidate parameters in Section 5.3, and we show how to get confidence regions in Section 5.4. Section 5.5 illustrates the validity of our parameter tuning and confidence regions with numerical experiments on smooth and noisy data.

5.1 Approximations of (MultiNerve) Mappers and Reeb graphs

In this section, we discuss the approximation of the Reeb graph, the (MultiNerve) Mapper and their signatures when the pair (X, f) is known only through a finite set of sample points equipped with function values.

Convention. From now on, we assume that the underlying space X is a compact Riemannian manifold of $\mathbb{R}^D$, $f : X \to \mathbb{R}$ is a Morse-type function and $X_n = \{x_1, \cdots, x_n\} \subset X$ is a point cloud in X of cardinality $n \in \mathbb{N}$. Moreover, we let $\|\cdot\|$ denote the usual Euclidean norm in $\mathbb{R}^D$. We often call f a *filter* function, as in the literature on applications of Reeb graph and Mappers.

5.1.1 Approximation tools

Rips complex. All constructions take a neighborhood graph as input, such as for instance the 1-skeleton graph of the Rips complex, defined as follows:

Definition 5.1.1. *Let $\delta \geq 0$ be a scale parameter. The* Rips complex *of X_n of parameter δ is the simplicial complex* $\mathrm{Rips}_\delta(X_n)$ *defined by:*

$$\{x_{i_0}, ..., x_{i_k}\} \in \mathrm{Rips}_\delta(X_n) \Leftrightarrow \|x_{i_p} - x_{i_q}\| \leq \delta \text{ for any } 0 \leq p, q \leq k, .$$

Its 1-skeleton graph is called the Rips graph *of parameter δ and denoted by* $\mathrm{Rips}^1_\delta(X_n)$.

Moreover, given a geometric realization $|\mathrm{Rips}_\delta(X_n)|$, we let $f^{\mathrm{PL}} : |\mathrm{Rips}_\delta(X_n)| \to \mathbb{R}$ denote the piecewise-linear interpolation of f along the simplices of $\mathrm{Rips}_\delta(X_n)$*, and we let* $\mathrm{R}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|)$ *denote its Reeb graph, with induced function $\widetilde{f}^{\mathrm{PL}}$.*

Geometric quantities. Two geometric quantities that assess the smoothness of topological spaces will be used in the hypotheses of the results in this chapter—see Theorem 5.1.5 and Theorem 5.2.1.

- *Reach.* The *medial axis* of $X \subset \mathbb{R}^D$ is the set of points in $\mathbb{R}^D$ with at least two nearest neighbors in X:
$$\mathrm{med}(X) = \{y \in \mathbb{R}^D \,:\, \mathrm{card}\{x \in X \,:\, \|y - x\| = \|y, X\|\} \geq 2\},$$
where $\|y, X\| = \inf\{\|y - x\| \,:\, x \in X\}$. The *reach* of X, denoted by $\mathrm{rch}(X)$, is the distance of X to its medial axis:
$$\mathrm{rch}(X) = \inf\{\|x - m\| \,:\, x \in X, m \in \mathrm{med}(X)\}.$$

- *Convexity radius.* A set $Y \subseteq X$ is said to be *convex* whenever every geodesic path in X between two points of Y stays in Y. The *convexity radius* of X is the smallest radius ρ for which every geodesic ball in X of radius less than ρ is convex.

Regularity of the filter function. Intuitively, approximating a Reeb graph computed with a filter function f that has large variations is more difficult than for a smooth filter function, for some notion of regularity that we now specify. Our result is given in a general setting by considering the *modulus of continuity of f*.

Definition 5.1.2. *Let $f : X \to \mathbb{R}$ be a Morse-type function. The* modulus of continuity *ω_f of f is:*

$$\omega_f : \left\{ \begin{array}{lcl} \mathbb{R}_+ & \to & \mathbb{R}_+ \\ \delta & \mapsto & \sup\{|f(x) - f(x')| \,:\, x, x' \in X,\ \|x - x'\| \leq \delta\} \end{array} \right.$$

It follows from the Definition 5.1.2 that ω_f satisfies :

1. $\omega_f(\delta) \to \omega(0) = 0$ as $\delta \to 0$;
2. ω_f is nonnegative and non-decreasing on $\mathbb{R}^+$;
3. ω_f is subadditive: $\omega_f(\delta_1 + \delta_2) \leq \omega_f(\delta_1) + \omega_f(\delta_2)$ for any δ_1, $\delta_2 > 0$;
4. ω_f is continous on $\mathbb{R}^+$.

Modulus of continuity. More generally, we say that a function ω defined on $\mathbb{R}_+$ is *a modulus of continuity* if it satisfies the four properties above, and we say that it is *a modulus of continuity for f* if, in addition, we have $\omega \geq \omega_f$.

5.1.2 Discrete approximations

Reeb graph

The following theorem states that the Rips complex of a point cloud can be used as a proxy for the original space X. Hence, it is possible to approximate the Reeb graph of X, in the bottleneck distance, by computing the Reeb graph of the Rips complex built on top of the point cloud.

Theorem 5.1.3 (Theorem 4.6 and Remark 2 in [64])**.** *Assume X has reach* $\mathrm{rch} > 0$ *and convexity radius $\rho > 0$. Let $\delta \geq 0$ be a scale parameter, and let ω be a modulus of continuity for f.*

If $4d_{\mathrm{H}}(X, X_n) \leq \delta \leq \min\left\{\frac{1}{4}\mathrm{rch}, \frac{1}{4}\rho\right\}$, then:

$$d_{\mathrm{b}}(\mathrm{Ext}_1^-(\tilde{f}), \mathrm{Ext}_1^-(\tilde{f}^{\mathrm{PL}})) \leq 2\omega(\delta).$$

Note that the original version of this theorem is only proven for Lipschitz functions in [64], but it extends at no cost, i.e. with the same proof, to functions with modulus of continuity.

Theorem 5.1.4 (Theorem 2 in [50])**.** *Assume X has convexity radius $\rho > 0$. Let $\delta > 0$ be a scale parameter, and let ω be a modulus of continuity for f.*

If $4d_{\mathrm{H}}(X, X_n) \leq \delta \leq \rho$, then:

$$\max\{d_{\mathrm{b}}(\mathrm{Ord}_0(\tilde{f}), \mathrm{Ord}_0(\tilde{f}^{\mathrm{PL}})), d_{\mathrm{b}}(\mathrm{Ext}_0^+(\tilde{f}), \mathrm{Ext}_0^+(\tilde{f}^{\mathrm{PL}})), d_{\mathrm{b}}(\mathrm{Rel}_1(\tilde{f}), \mathrm{Rel}_1(\tilde{f}^{\mathrm{PL}}))\} \leq \omega(\delta).$$

Again, the original version of this theorem is only proven for Lipschitz functions in [50], but it extends at no cost to functions with modulus of continuity. Moreover, three more remarks need to be made. Firstly, this theorem is originally stated only for the ordinary part of the persistence diagrams but its proof extends to the full extended filtrations at no extra cost. Secondly, it is stated for a nested pair of Rips complexes, however, as pointed out in Section 4.3 in [50], in 0-dimensional homology a single Rips graph is sufficient for the theorem to hold. Thirdly, its approximation function is piecewise-constant and not piecewise-linear as in this article. However, the filtrations induced by the sublevel sets and the superlevel sets of the piecewise-constant function are actually lower- and upper-star filtrations, and it is known in that case that piecewise-linear and piecewise-constant functions induce the same persistence diagram. See Section 2.5 of [103] for a proof of this statement.

Combining the two theorems gives the following complete approximation result:

Theorem 5.1.5. *Under the assumptions of Theorem 5.1.3, we have* $d_{\mathrm{b}}(\mathrm{ExDg}(\tilde{f}), \mathrm{ExDg}(\tilde{f}^{\mathrm{PL}})) \leq 2\omega(\delta)$.

(MultiNerve) Mapper

We report three possible constructions for the (MultiNerve) Mapper from the pair (X_n, f): the first is from the original Mapper paper [129], the second is inspired from the graph-induced complex paper [63], and the third is simply the Rips complex approximation. Given a choice of neighborhood parameter δ and the corresponding Rips graph, the construction from [129] uses the vertices as witnesses for the connected components of the pullback cover on $\mathrm{Rips}_\delta^1(X_n)$ and for their pairwise intersections. Differently, the construction from [63] uses the edges as witnesses for the pairwise intersections. Thus, both constructions have the same vertex set but potentially different edge sets.

Vertex-based connectivity. Given an arbitrary interval I in $\mathbb{R}$, the preimage of I in X_n is defined to be $X_n \cap f^{-1}(I)$, and its connected components are defined to be the connected components of the induced subgraph $\mathrm{Rips}_\delta^1(X_n \cap f^{-1}(I))$. Then, the vertices in the (MultiNerve) Mapper are the connected components of the preimages of the intervals $I \in \mathcal{I}$. Given two intersecting intervals I, J of $\mathcal{I}$, given a connected component C_I in the preimage of I and a connected component C_J in the preimage of J, the corresponding vertices are connected by an edge in the Mapper if there is a connected component in the preimage of $I \cap J$ that is contained in both C_I and C_J; in the MultiNerve Mapper, there are as many copies of this edge as there are connected components in the preimage of $I \cap J$ that are contained in $C_I \cap C_J$. We denote these two constructions by $\mathrm{M}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), \mathcal{I})$ and $\overline{\mathrm{M}}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), \mathcal{I})$ respectively. Moreover, functions $\mathsf{m}_{\mathcal{I}}^\bullet$ and $\bar{\mathsf{m}}_{\mathcal{I}}^\bullet$ can be defined on these constructions exactly like in Definition 4.3.1.

Edge-based connectivity. The vertex set of the (MultiNerve) Mapper is the same as in the previous construction. Now, for any intersecting intervals I, J of $\mathcal{I}$, we redefine the preimage of the intersection $I \cap J$ to be the subset of $\mathrm{Rips}^1_\delta(X_n)$ spanned not only by the points of $X_n \cap f^{-1}(I \cap J)$ and the graph edges connecting them, but also by the relative interiors of the edges of $\mathrm{Rips}^1_\delta(X_n)$ that have one vertex in $X_n \cap f^{-1}(I)$ and the other in $X_n \cap f^{-1}(J)$. Then, given a connected component C_I in the preimage of I and a connected component C_J in the preimage of J, we connect the corresponding vertices by an edge in the Mapper if there is a connected component of the redefined preimage of $I \cap J$ that connects[1] C_I and C_J in $\mathrm{Rips}^1_\delta(X_n)$; in the MultiNerve Mapper, we add as many copies of this edge as there are connected components in the redefined preimage of $I \cap J$ that connect C_I and C_J. We denote these two constructions by $\mathrm{M}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I})$ and $\overline{\mathrm{M}}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I})$ respectively. Again, we also define functions $\mathsf{m}^{\triangle}_{\mathcal{I}}$ and $\bar{\mathsf{m}}^{\triangle}_{\mathcal{I}}$ on these constructions.

Rips complex. As for Reeb graphs, one can compute (MultiNerve) Mappers $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, \mathcal{I})$ and $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, \mathcal{I})$ from a geometric realization of a Rips complex built on top of X_n with parameter δ, using the piecewise-linear extension f^{PL}. Seeing $|\mathrm{Rips}_\delta(X_n)|$ as a topological space, let also $\bar{T}_{\mathcal{I}}(|\mathrm{Rips}_\delta(X_n)|, f^{\mathrm{PL}})$ and $T_{\mathcal{I}}(|\mathrm{Rips}_\delta(X_n)|, f^{\mathrm{PL}})$ denote the telescopes obtained with Definition 4.6.2 and Definition 4.6.7, with corresponding projections onto the second factor $\bar{f}^{\mathrm{PL}}_{\mathcal{I}}$ and $f^{\mathrm{PL}}_{\mathcal{I}}$. We recall that $\mathcal{C}\mathrm{R}_{\bar{f}^{\mathrm{PL}}_{\mathcal{I}}}(\bar{T}_{\mathcal{I}}(|\mathrm{Rips}_\delta(X_n)|, f^{\mathrm{PL}}))$ and $\mathcal{C}\mathrm{R}_{f^{\mathrm{PL}}_{\mathcal{I}}}(T_{\mathcal{I}}(|\mathrm{Rips}_\delta(X_n)|, f^{\mathrm{PL}}))$ are isomorphic to $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, \mathcal{I})$ and $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, \mathcal{I})$ respectively, according to Theorem 4.6.4 and Corollary 4.6.8. Moreover, the induced maps $\tilde{\bar{f}}^{\mathrm{PL}}_{\mathcal{I}} = \bar{\mathsf{m}}^{\mathrm{PL}}_{\mathcal{I}}$ and $\tilde{f}^{\mathrm{PL}}_{\mathcal{I}} = \mathsf{m}^{\mathrm{PL}}_{\mathcal{I}}$ are related to $\tilde{f}^{\mathrm{PL}}$ according to Theorem 4.6.12. See Figure 5.1 for an illustration of all functions defined on Mappers considered here.

5.1.3 Relationships between the constructions

In each of the three constructions detailed above, the Mapper is included in the MultiNerve Mapper by definition. Moreover, the preimages of the intersections in the second construction are supersets of the preimages in the first construction, and two different connected components in the same preimage in the first construction cannot be connected in the second construction, therefore the (MultiNerve) Mapper from the first construction is included in its counterpart from the second construction. Hence the following diagram of inclusions:

$$\begin{tikzcd} \mathrm{M}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I}) \arrow[r] \arrow[d] & \overline{\mathrm{M}}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I}) \arrow[d] \\ \mathrm{M}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I}) \arrow[r] & \overline{\mathrm{M}}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I}) \end{tikzcd} \tag{5.1}$$

The vertical inclusions become equalities when there are no *intersection-crossing edges* in the Rips graph, defined as follows:

Definition 5.1.6. *An edge $[u, v]$ of the Rips graph is* interval-crossing *if there is an interval $I \in \mathcal{I}$ such that $I \subseteq (\min\{f(u), f(v)\}, \max\{f(u), f(v)\})$. It is* intersection-crossing *if there is a pair of intervals $I, J \in \mathcal{I}$ such that $\emptyset \neq I \cap J \subseteq (\min\{f(u), f(v)\}, \max\{f(u), f(v)\})$.*

Indeed, in the absence of intersection-crossing edges, each connected component in the preimage of an interval intersection in the second construction contains a vertex and therefore deform retracts onto the corresponding connected component in the first construction. Hence:

Lemma 5.1.7. *If there are no intersection-crossing edges in $\mathrm{Rips}^1_\delta(X_n)$, then $\mathrm{M}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I})$ is isomorphic to $\mathrm{M}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I})$ as combinatorial multigraphs. The same is true for $\overline{\mathrm{M}}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I})$ and $\overline{\mathrm{M}}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n), \mathcal{I})$.*

[1] By which we mean that the closure of the connected component in $\mathrm{Rips}^1_\delta(X_n)$ contains points from C^I and from C^J.

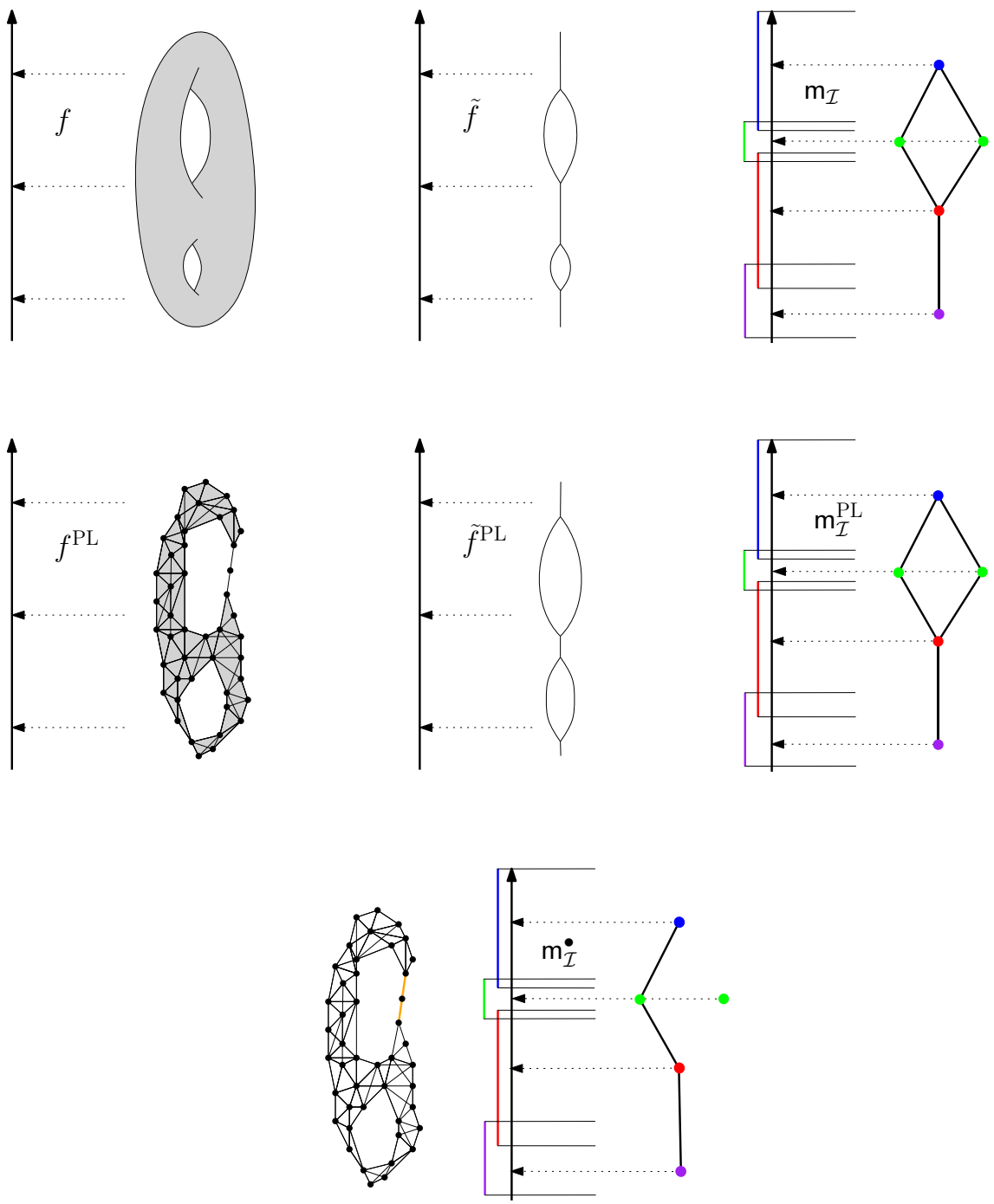

Figure 5.1: Examples of the function defined on the original space (left column), its induced function defined on the Reeb graph (middle column) and the function defined on the Mapper (right column). Note that the Mapper computed from the geometric realization of the Rips complex (middle row, right) is not isomorphic to the standard Mapper (last row), since there are two intersection-crossing edges in the Rips complex (outlined in orange).

Concerning the relation between the first two constructions and the third one, we have:

Lemma 5.1.8. *If there are no interval-crossing edges in* $\mathrm{Rips}^1_\delta(X_n)$*, then* $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$ *is isomorphic to* $\mathrm{M}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})$ *as combinatorial multigraphs. The same is true for* $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$ *and* $\overline{\mathrm{M}}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})$*.*

Proof. Note that $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$ and $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$ are the same as $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}^1_\delta(X_n)|,\mathcal{I})$ and $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}^1_\delta(X_n)|,\mathcal{I})$ respectively, since only the connected component of the preimages of intervals are involved in the construction of the (MultiNerve) Mapper. Hence, for the rest of the proof we set the domain of f^{PL} to be $|\mathrm{Rips}^1_\delta(X_n)|$. Every connected component in the preimage through f^{PL} of an interval of $\mathcal{I}$ must contain a vertex, therefore it deform retracts onto the corresponding preimage through f. Hence the vertex sets of the aforementioned simplicial posets are the same. Every connected component in the preimage through f^{PL} of an interval intersection $I\cap J$ either contains a vertex, in which case it deform retracts onto the corresponding preimage through f in the vertex-based connectivity, or it does not contain any vertex, in which case the edge of the Rips graph that contains the connected component creates an edge in the (MultiNerve) Mapper in the edge-based connectivity. □

See Figure 5.2 for an example showing the importance of the hypothesis in the lemma.

5.1.4 Relationships between the signatures

Relationships between the (MultiNerve) Mapper constructions. The following diagram summarizes the relationships between the various (MultiNerve) Mapper constructions:

$$\begin{tikzcd}
(X,f) \arrow[d, leftrightarrow] & (|\mathrm{Rips}_\delta(X_n)|), f^{\mathrm{PL}}) \arrow[d, leftrightarrow] & & \\
(\mathrm{R}_f(X),\tilde{f}) \arrow[r, dashed, leftrightarrow] \arrow[d, leftrightarrow] & (\mathrm{R}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|),\tilde{f}^{\mathrm{PL}}) \arrow[d, leftrightarrow] & & \\
\overline{\mathrm{M}}_f(X,\mathcal{I}) \arrow[d, leftrightarrow] & \overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I}) \arrow[r, leftrightarrow, dotted] \arrow[d, leftrightarrow] & \overline{\mathrm{M}}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I}) \arrow[r, leftrightarrow, dotted] \arrow[d, leftrightarrow] & \overline{\mathrm{M}}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I}) \arrow[d, leftrightarrow] \\
\mathrm{M}_f(X,\mathcal{I}) & \mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I}) \arrow[r, leftrightarrow, dotted] & \mathrm{M}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I}) \arrow[r, leftrightarrow, dotted] & \mathrm{M}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})
\end{tikzcd} \tag{5.2}$$

The vertical arrows between the first and second rows are provided by Theorem 2.4.4. The ones between the second, third and fourth rows are given by Eqs. (4.2) and (4.3). The dotted horizontal arrows are provided by Lemmas 5.1.7 and 5.1.8. Finally, the dashed horizontal arrow is given by Theorem 5.1.5.

Approximation of the (MultiNerve) Mapper. We then derive from (5.2) the following approximation guarantee:

Theorem 5.1.9. *Under the assumptions of Theorem 5.1.3, and given a gomic* $\mathcal{I}$*, we have:*

$$d_{\mathcal{I}}\left(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\, \mathrm{ExDg}(\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I}))\right) \leq 2\omega(\delta).$$

If furthermore there are no interval-crossing edges, then $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$ *and* $\overline{\mathrm{M}}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})$ *are isomorphic as combinatorial multigraphs. If there are no intersection-crossing edges either, then* $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$ *and* $\overline{\mathrm{M}}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})$ *are also isomorphic as combinatorial multigraphs.*

The same result holds for $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|,\mathcal{I})$, $\mathrm{M}^{\triangle}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})$ and $\mathrm{M}^{\bullet}_f(\mathrm{Rips}^1_\delta(X_n),\mathcal{I})$, provided $d_{\mathcal{I}}$ is replaced by the bottleneck distance with the appropriate extended staircase $Q^{\mathcal{I}}_E$. Thus, we can construct discrete (MultiNerve) Mappers whose signatures approximate the ones of the corresponding continuous structures $\mathrm{M}_f(X,\mathcal{I})$ and $\overline{\mathrm{M}}_f(X,\mathcal{I})$.

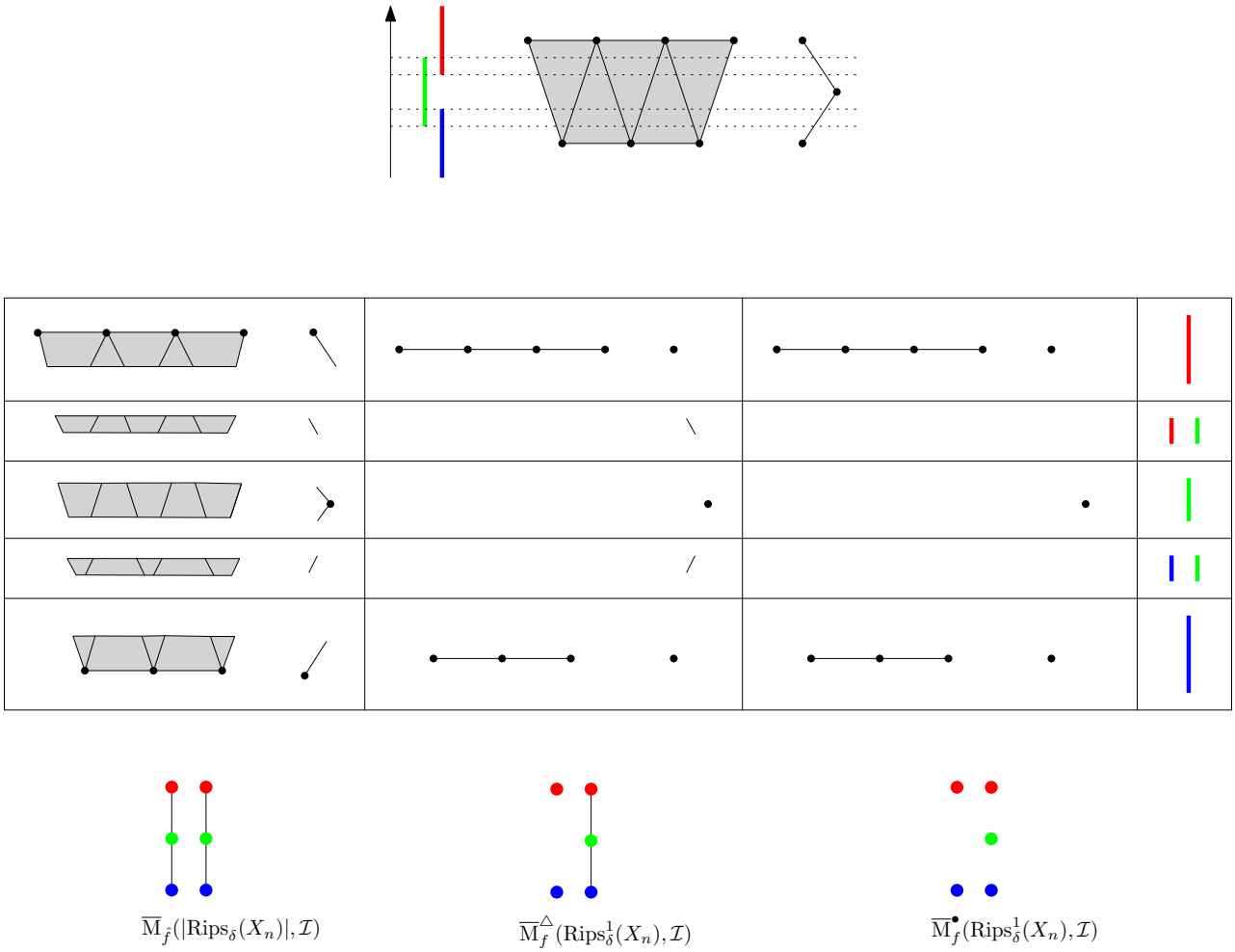

Figure 5.2: We study a Rips complex $\mathrm{Rips}_{\delta}(X_n)$ built on top of a point cloud X_n with ten points. This complex has two connected components. We also compute (MultiNerve) Mappers with the height function, whose image is covered by three intervals. We display the preimages of the intervals and their intersections for $\overline{\mathrm{M}}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_{\delta}(X_n)|,\mathcal{I})$, $\overline{\mathrm{M}}_{f}^{\triangle}(\mathrm{Rips}_{\delta}^{1}(X_n),\mathcal{I})$ and $\overline{\mathrm{M}}_{f}^{\bullet}(\mathrm{Rips}_{\delta}^{1}(X_n),\mathcal{I})$. The edges of the right connected component are intersection-crossing but not interval-crossing, so $\overline{\mathrm{M}}_{f}^{\triangle}(\mathrm{Rips}_{\delta}^{1}(X_n),\mathcal{I})$ recovers it correctly while $\overline{\mathrm{M}}_{f}^{\bullet}(\mathrm{Rips}_{\delta}^{1}(X_n),\mathcal{I})$ fails to. The edges of the left connected component are interval-crossing, so both $\overline{\mathrm{M}}_{f}^{\bullet}(\mathrm{Rips}_{\delta}^{1}(X_n),\mathcal{I})$ and $\overline{\mathrm{M}}_{f}^{\triangle}(\mathrm{Rips}_{\delta}^{1}(X_n),\mathcal{I})$ fail to recover the connected component.

Approximation of the (MultiNerve) Mapper signature. In some situations, one is merely interested in approximating the signatures of $\mathrm{M}_f(X,\mathcal{I})$ and $\overline{\mathrm{M}}_f(X,\mathcal{I})$ without actually building corresponding discrete (MultiNerve) Mappers. In such cases, one can simply apply the scalar fields analysis approach of [42] to approximate $\mathrm{ExDg}(f)$, then remove the points from $(\mathrm{Ext}_1^+(f)\cup\mathrm{Ord}_1(f))$ as well as the points lying in their corresponding staircases, to get an approximation of the signatures:

Theorem 5.1.10. *Under the assumptions of Theorem 5.1.3, and given a gomic $\mathcal{I}$, let* ExDg *denote the extended persistence diagram computed by the algorithm of [50], and then pruned by removing the points of the* Ext_1^+ *and* Ord_1 *subdiagrams as well as the points located in the staircase corresponding to their type. Then this diagram approximates the signature of* $\overline{\mathrm{M}}_f(X,\mathcal{I})$ *as follows:*

$$d_{\mathcal{I}}\left(\mathrm{ExDg}(\overline{\mathrm{M}}_f(X,\mathcal{I})),\ \mathrm{ExDg}\right)\leq 2\omega(\delta).$$

The same bound applies for the approximation of $\mathrm{ExDg}(\mathrm{M}_f(X,\mathcal{I}))$*, provided the staircase* $Q_{E^-}^{\mathcal{I}}$ *is replaced by its extended version* $Q_{E}^{\mathcal{I}}$ *in the definitions of* $d_{\mathcal{I}}$ *and* ExDg.

Note that this result holds much more generally than Theorem 5.1.9, however there may be no discrete (MultiNerve) Mapper construction associated with the approximate diagram ExDg.

5.2 Approximation of a Reeb graph with Mapper

In this section, we state and prove Theorem 5.2.1. This result states that the vertex-based Mapper $\mathrm{M}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), \mathcal{I})$, which is the one that is used by most practitioners, is an approximation of the Reeb graph in the bottleneck distance d_b. However, two remarks have to be made at this stage.

- First, all Mappers in this chapter are computed on a point cloud X_n, whereas they are computed on the support $X \supset X_n$ in Chapter 4. In particular, the signature (4.3) is not well-defined, so we rather use $\mathrm{ExDg}(\mathsf{m}_\mathcal{I}^\bullet)$ as a signature when computing the bottleneck distance.
- Second, we cannot use our previous approximation results (Theorems 5.1.9 and 5.1.10), since they are stated with $d_\mathcal{I}$. Indeed, even though $d_\mathcal{I}$ is a natural distance stabilizing Mappers, it is not suited for Reeb graphs since they do not depend on gomics.

Convention. We use gomics $\mathcal{I}$ of $\mathrm{im}(f)$ whose elements have constant length $r > 0$ (apart from the first and last one, which can have any positive length) and constant overlap percentage $g \in \left(0, \frac{1}{2}\right)$, i.e. $\mathrm{length}(I \cap J) = gr$, for any consecutive intervals I, J in $\mathcal{I}$. The parameter r is called the *resolution* of $\mathcal{I}$, and the parameter g is called its *gain*. We often write (r, g) instead of $\mathcal{I}$.

Theorem 5.2.1. *Assume X has reach* $\mathrm{rch} > 0$ *and convexity radius $\rho > 0$. Assume that the filter function f is Morse-type on X. Let ω be a modulus of continuity for f. If the three following conditions on parameter δ hold:*

$$\delta \leq \min\left\{\frac{1}{4}\mathrm{rch}, \frac{1}{4}\rho\right\}, \tag{5.3}$$

$$\max\{|f(x) - f(x')| \,:\, x, x' \in X_n, \|x - x'\| \leq \delta\} < gr, \tag{5.4}$$

$$4d_\mathrm{H}(X, X_n) \leq \delta, \tag{5.5}$$

then the Mapper $\mathrm{M}_n = \mathrm{M}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), (r, g))$ *is such that:*

$$d_\mathrm{b}\left(\mathrm{R}_f(X), \mathrm{M}_n\right) \leq r + 2\omega(\delta). \tag{5.6}$$

Remark 5.2.2. *Using the edge-based version* $\mathrm{M}_f^\triangle(\mathrm{Rips}_\delta^1(X_n), \mathcal{I})$ *allows to weaken Assumption (5.4) since gr can be replaced by r in the corresponding equation. In addition, using the MultiNerve Mapper instead of the Mapper allows to replace r by $r/2$ in Equation (5.6).*

Proof of Theorem 5.2.1. The following inequalities lead to the result:

$$d_\mathrm{b}(\mathrm{R}_f(X), \mathrm{M}_n) = d_\mathrm{b}\left(\mathrm{ExDg}(\tilde{f}), \mathrm{ExDg}(\mathsf{m}_\mathcal{I}^\bullet)\right) = d_\mathrm{b}\left(\mathrm{ExDg}(\tilde{f}), \mathrm{ExDg}(\mathsf{m}_\mathcal{I}^\mathrm{PL})\right) \tag{5.7}$$

$$\leq d_\mathrm{b}\left(\mathrm{ExDg}(\tilde{f}), \mathrm{ExDg}(\tilde{f}^\mathrm{PL})\right) + d_\mathrm{b}\left(\mathrm{ExDg}(\tilde{f}^\mathrm{PL}), \mathrm{ExDg}(\mathsf{m}_\mathcal{I}^\mathrm{PL})\right) \tag{5.8}$$

$$\leq 2\omega(\delta) + r. \tag{5.9}$$

Let us prove every (in)equality.

Equality (5.7). The first equality is the definition of the bottleneck distance for graphs.

To prove the second equality, we have to show that M_n and $\mathrm{M}_{f^\mathrm{PL}}(|\mathrm{Rips}_\delta(X_n)|, (r, g))$ are isomorphic. Let $x_1, x_2 \in X_n$ such that (x_1, x_2) is an edge of $\mathrm{Rips}_\delta^1(X_n)$ i.e. $\|x_1 - x_2\| \leq \delta$. Then, according to (5.4): $|f(x_1) - f(x_2)| \leq gr$. Hence, there is no $\alpha \in \{1, \ldots, \mathrm{card}(\mathcal{I}) - 1\}$ such that $I_\alpha \cap I_{\alpha+1} \subseteq (\min\{f(x_1), f(x_2)\}, \max\{f(x_1), f(x_2)\})$. It follows that there are no intersection-crossing and interval-crossing edges in $\mathrm{Rips}_\delta^1(X_n)$. Then, according to Lemma 5.1.7 and Lemma 5.1.8, there is a graph isomorphism $i : \mathrm{M}_n = \mathrm{M}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), (r, g)) \to \mathrm{M}_{f^\mathrm{PL}}(|\mathrm{Rips}_\delta(X_n)|, (r, g))$. Since $\mathsf{m}_\mathcal{I}^\bullet = \mathsf{m}_\mathcal{I}^\mathrm{PL} \circ i$ by definition of $\mathsf{m}_\mathcal{I}^\bullet$ and $\mathsf{m}_\mathcal{I}^\mathrm{PL}$, the equality follows.

Inequality (5.8). This inequality is just an application of the triangle inequality.

Inequality (5.9). According to (5.3), we have $\delta \leq \min\{\frac{1}{4}\mathrm{rch}, \frac{1}{4}\rho\}$. According to (5.5), we also have $\delta \geq 4d_{\mathrm{H}}(X, X_n)$. Hence, we have

$$d_{\mathrm{b}}(\mathrm{ExDg}(\tilde{f}), \mathrm{ExDg}(\tilde{f}^{\mathrm{PL}})) \leq 2\omega(\delta),$$

according to Theorem 5.1.5. Moreover, we have

$$d_{\mathrm{b}}(\mathrm{ExDg}(\tilde{f}^{\mathrm{PL}}), \mathrm{ExDg}(\mathsf{m}_{\mathcal{I}}^{\mathrm{PL}})) \leq r,$$

according to Theorem 4.6.12.

□

Analysis of the hypotheses. On the one hand, the scale parameter of the Rips complex could not be smaller than the approximation error corresponding to the Hausdorff distance between the sample X_n and the underlying space X (Assumption (5.5)). On the other hand, it must be smaller than the reach and convexity radius to provide a correct estimation of the geometry and topology of X (Assumption (5.3)). The quantity gr corresponds to the minimum scale at which the filter's codomain is analyzed. This minimum resolution has to be compared with the regularity of the filter at scale δ (Assumption (5.4)). Indeed the pre-images of a filter with strong variations will be more difficult to analyze than when the filter varies little.

Analysis of the upper bound. The upper bound given in (5.6) makes sense in that the approximation error is controlled by the resolution level in the codomain and by the regularity of the filter. If one uses a filter with strong variations, or if the grid in the codomain has a too rough resolution, then the approximation will be poor. On the other hand, a sufficiently dense sampling is required in order to take r small, as prescribed in the assumptions.

Lipschitz filters. A large class of filters used for Mapper are actually Lipschitz functions and of course, in this case, one can take $\omega(\delta) = c\delta$ for some positive constant c. In particular, $c = 1$ for linear projections (PCA, SVD, Laplacian or coordinate filter for instance). The distance to a measure (DTM) is also a 1-Lipschitz function, see [44]. On the other hand, the modulus of continuity of filter functions defined from estimators, e.g. density estimators, is less obvious although still well-defined.

Filter approximation. In some situations, the filter function $\hat{f}$ used to compute the Mapper is only an approximation of the filter function f with which the Reeb graph is computed. In this context, the pair $(X_n, \hat{f})$ appears as an approximation of the pair (X, f). The following result is directly derived from Theorem 5.2.1 and Theorem 4.4.2.

Corollary 5.2.3. *Let $\hat{f} : X \to \mathbb{R}$ be a Morse-type filter function approximating f. Suppose that Assumptions (5.3) to (5.5) of Theorem 5.2.1 are satisfied by both f and $\hat{f}$. Then, the Mapper $\hat{\mathrm{M}}_n = \mathrm{M}^{\bullet}_{\hat{f}}(\mathrm{Rips}^1_\delta(X_n), (r, g))$ built on X_n with filter function $\hat{f}$, satisfies:*

$$d_{\mathrm{b}}(\mathrm{R}_f(X), \hat{\mathrm{M}}_n) \leq 2r + 2\omega(\delta) + \max\{|f(x_i) - \hat{f}(x_i)| \,:\, 1 \leq i \leq n\}.$$

Proof. Let $\hat{f}^{\mathrm{PL}}$ be the piecewise-linear interpolation of $\hat{f}$ on the simplices of $\mathrm{Rips}^1_\delta(X_n)$. As before, since $|\mathrm{Rips}_\delta(X_n)|$ and $|\mathrm{Rips}_\delta(X_n)|$ are metric spaces, we also consider their Mappers $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r, g))$ and $\mathrm{M}_{\hat{f}^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r, g))$. Then, the following inequalities lead to the result:

$$\begin{aligned}
d_{\mathrm{b}}(\mathrm{R}_f(X), \hat{\mathrm{M}}_n) &\leq d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{M}_n) + d_{\mathrm{b}}(\mathrm{M}_n, \hat{\mathrm{M}}_n) \text{ by the triangle inequality} \\
&= d_{\mathrm{b}}(\mathrm{R}_f(X), \mathrm{M}_n) + d_{\mathrm{b}}(\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r, g)), \mathrm{M}_{\hat{f}^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r, g))) \qquad (5.10) \\
&\leq r + 2\omega(\delta) + d_{\mathrm{b}}(\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r, g)), \mathrm{M}_{\hat{f}^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r, g))) \text{ by Theorem 5.2.1} \\
&\leq r + 2\omega(\delta) + r + \|f^{\mathrm{PL}} - \hat{f}^{\mathrm{PL}}\|_\infty \text{ by Theorem 4.4.2} \\
&= 2r + 2\omega(\delta) + \max\{|f(x) - \hat{f}(x)| \,:\, x \in X_n\}
\end{aligned}$$

Let us prove Equality (5.10). Since f and $\hat{f}$ satisfy Assumption (5.4), there are no intersection-crossing edges for both f and $\hat{f}$. According to Lemma 5.1.7 and Lemma 5.1.8, $\mathrm{M}_{f^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r,g))$ and M_n are isomorphic and similarly for $\mathrm{M}_{\hat{f}^{\mathrm{PL}}}(|\mathrm{Rips}_\delta(X_n)|, (r,g))$ and $\hat{\mathrm{M}}_n$. See also the proof of Equality (5.7).

□

5.3 Statistical Analysis of the Mapper

In this section, we study the rates of convergence of the discrete Mapper $\mathrm{M}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), (r,g))$. We first show that the Mapper is a measurable construction (Proposition 5.3.1).

Convention. From now on, the set of observations X_n is assumed to be composed of n independent points $x_1, ..., x_n$ sampled from a probability distribution $\mathbb{P}$ in $\mathbb{R}^D$ (endowed with its Borel algebra). We assume that each point x_i comes with a filter function value which is represented by a random variable $y_i = f(x_i)$. Contrarily to the x_i's, the filter values y_i's are not necessarily independent. We use $\mathrm{M}_{r,g,\delta}(X_n, Y_n)$ as a shorthand for $\mathrm{M}_f^\bullet(\mathrm{Rips}_\delta^1(X_n), (r,g))$ to emphasize the separation between random variables and Mapper parameters.

Measurability of the Mapper. We first provide the following proposition, which states that computing probabilities on the Mapper makes sense:

Proposition 5.3.1. *For any fixed choice of parameters r, g, δ and for any fixed $n \in \mathbb{N}$, the function*

$$\Phi : \left\{ \begin{array}{ccc} (\mathbb{R}^D)^n \times \mathbb{R}^n & \to & \mathsf{Reeb} \\ (X_n, Y_n) & \mapsto & \mathrm{M}_{r,g,\delta}(X_n, Y_n) \end{array} \right.$$

is measurable.

We recall that $\mathrm{M}_{r,g,\delta}(X_n, Y_n) \in \mathsf{Reeb}$ according to Definition 4.6.2 and Theorem 4.6.4.

Proof. We check that not only the topological signature of the Mapper but also the Mapper itself is a measurable object and thus can be seen as an estimator of a target Reeb graph. This problem is more complicated than for the statistical framework of persistence diagram inference, for which the existing stability results give for free that persistence estimators are measurable for adequate sigma algebras.

Let $\bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$ denote the extended real line. Given a fixed integer $n \geq 1$, let $\mathcal{C}_{[n]}$ be the set of abstract simplicial complexes over a fixed set of n vertices. We see $\mathcal{C}_{[n]}$ as a subset of the power set $2^{2^{[n]}}$, where $[n] = \{1, \cdots, n\}$, and we implicitly identify $2^{[n]}$ with the set $[2^n]$ via the map assigning to each subset $\{i_1, \cdots, i_k\}$ the integer $1 + \sum_{j=1}^k 2^{i_j - 1}$. Given a fixed parameter $\delta > 0$, we define the application

$$\Phi_1 : \left\{ \begin{array}{ccc} (\mathbb{R}^D)^n \times \mathbb{R}^n & \to & \mathcal{C}_{[n]} \times \bar{\mathbb{R}}^{2^{[n]}} \\ (X_n, Y_n) & \mapsto & (K, f_K) \end{array} \right.$$

where K is the abstract Rips complex of parameter δ over the n labeled points in $\mathbb{R}^D$, minus the intersection-crossing edges and their cofaces, and where f_K is a function defined by:

$$f_K : \left\{ \begin{array}{ccc} 2^{[n]} & \to & \bar{\mathbb{R}} \\ \sigma & \mapsto & \left\{ \begin{array}{ll} \max_{i \in \sigma} Y_i & \text{if } \sigma \in K \\ +\infty & \text{otherwise.} \end{array} \right. \end{array} \right.$$

The space $(\mathbb{R}^D)^n \times \mathbb{R}^n$ is equipped with the standard topology, denoted by T_1, inherited from $\mathbb{R}^{(D+1)n}$. The space $\mathcal{C}_{[n]} \times \bar{\mathbb{R}}^{2^{[n]}}$ is equipped with the product, denoted by T_2 hereafter, of the discrete topology on $\mathcal{C}_{[n]}$ and the topology induced by the extended distance $d(f,g) = \max\{|f(\sigma) - g(\sigma)| : \sigma \in 2^{[n]}, f(\sigma) \text{ or } g(\sigma) \neq +\infty\}$ on $\bar{\mathbb{R}}^{2^{[n]}}$. In particular, $K \neq K' \Rightarrow d(f_K, f_{K'}) = +\infty$.

Note that the map $(X_n, Y_n) \mapsto K$ is piecewise-constant, with jumps located at the hypersurfaces defined by $\|x_i - x_j\|^2 = \delta^2$ (for combinatorial changes in the Rips complex) or $y_i = \text{cst} \in \text{End}((r,g))$ (for changes in the set of intersection-crossing edges) in $(\mathbb{R}^D)^n \times \mathbb{R}^n$, where $\text{End}((r,g))$ denotes the set of endpoints of elements of the gomic (r,g). We can then define a finite measurable partition $(\mathcal{C}_\ell)_{\ell \in L}$ of $(\mathbb{R}^D)^n \times \mathbb{R}^n$ whose boundaries are included in these hypersurfaces, and such that $(X_n, Y_n) \mapsto K$ is constant over each set $\mathcal{C}_\ell$. As a byproduct, we have that $(X_n, Y_n) \mapsto f$ is continuous over each set $\mathcal{C}_\ell$.

We now define the operator

$$\Phi_2 : \begin{cases} \mathcal{C}_{[n]} \times \bar{\mathbb{R}}^{2^{[n]}} & \to & \mathcal{A} \\ (K, f) & \mapsto & (|K|, f^{\text{PL}}) \end{cases}$$

where $\mathcal{A}$ denotes the class of topological spaces filtered by Morse-type functions, and where f^{PL} is the piecewise-linear interpolation of f on the geometric realization $|K|$ of K. For a fixed simplicial complex K, the extended persistence diagram of the lower-star filtration induced by f and of the sublevel sets of f^{PL} are identical—see e.g. [103], therefore the map Φ_2 is distance-preserving (hence continuous) in the pseudometrics d_{b} on the domain and codomain. Since the topology T_2 on $\mathcal{C}_{[n]} \times \bar{\mathbb{R}}^{2^n}$ is a refinement[2] of the topology induced by d_{b}, the map Φ_2 is also continuous when $\mathcal{C}_{[n]} \times \bar{\mathbb{R}}^{2^{[n]}}$ is equipped with T_2.

Let now $\Phi_3 : \mathcal{A} \to \mathsf{Reeb}$ map each Morse-type pair (X, f) to its Mapper $\text{M}_f(X, \mathcal{I})$ using Definition 4.6.2, where $\mathcal{I} = (r, g)$ is the gomic induced by r and g. Note that, similarly to Φ_1, the map Φ_3 is piecewise-constant, since combinatorial changes in $\text{M}_f(X, \mathcal{I})$ are located at the regions $\text{Crit}(f) \cap \text{End}(\mathcal{I}) \neq \emptyset$. Hence, Φ_3 is measurable in the pseudometric d_{b}. Moreover, $\text{M}_{f^{\text{PL}}}(|K|, \mathcal{I})$ is isomorphic to $\text{M}_{r,g,\delta}(X_n, Y_n)$ by Lemma 5.1.7 and Lemma 5.1.8. since all intersection-crossing edges were removed in the construction of K. Hence, the map Φ defined by $\Phi = \Phi_3 \circ \Phi_2 \circ \Phi_1$ is a measurable map that sends (X_n, Y_n) to $\text{M}_{r,g,\delta}(X_n, Y_n)$. □

5.3.1 Statistical Model for the Mapper

Generative model. We now introduce the generative model for our data. Recall that the set of observations X_n is composed of n independent points $x_1, ..., x_n$ sampled from a probability distribution $\mathbb{P}$ in $\mathbb{R}^D$. The support of $\mathbb{P}$ is denoted $X_{\mathbb{P}}$ and is assumed to be a compact Riemannian manifold of $\mathbb{R}^D$ with positive reach and convexity radius, as in the setting of Theorem 5.2.1. We also assume that the diameter of $X_{\mathbb{P}}$ is bounded by some constant $L > 0$, i.e. we have $0 < \text{diam}(X_{\mathbb{P}}) \leq L$. Next, the probability distribution $\mathbb{P}$ is assumed to be (a, b)-standard for some constants $a > 0$ and $b \geq d$:

Definition 5.3.2. *Let $a, b > 0$. A probability distribution $\mathbb{P}$ is said to be (a, b)-standard if for any Euclidean ball $B(x, t)$ centered on $x \in X$ with radius t :*

$$\mathbb{P}(B(x, t)) \geq \min(1, at^b).$$

This assumption is popular in the literature about set estimation—see for instance [58, 59]. It is also widely used in the literature on persistence diagram estimation [47, 49, 71]. For instance, when $b = D$, this assumption is satisfied when the distribution is absolutely continuous with respect to the Hausdorff measure on $X_{\mathbb{P}}$. We introduce the set $\mathcal{P}_{a,b} = \mathcal{P}_{a,b,\kappa,\rho,L}$ which is composed of all the (a, b)-standard probability distributions $\mathbb{P}$ for which the support $X_{\mathbb{P}}$ is a compact Riemannian manifold of $\mathbb{R}^D$ with reach at least κ, convexity radius at least ρ and diameter at most L.

Filter functions in the statistical setting. The filter function $f : X_{\mathbb{P}} \to \mathbb{R}$ for the Reeb graph is assumed as before to be a Morse-type function. Two different settings have to be considered regarding how the filter function is defined. In the first setting, the same filter function is used to define the Reeb graph and the Mapper. The Mapper can be defined by taking the exact values of the filter function at the observation points $f(x_1), \ldots, f(x_n)$. Note that this does not mean that the function f is completely known since, in our framework, knowing f would imply to know its domain and thus $X_{\mathbb{P}}$ would be known which is of course not the case in practice. This first setting is studied in Section 5.3.2, and referred to as the *exact filter setting* in the following. It corresponds to the situations where the

[2] This is because singletons are open balls in the discrete topology, and also because of Theorem 2.3.1.

Mapper algorithm is used with coordinate functions for instance. In this setting, we distinguish two different cases corresponding to whether the parameters of the generative model are known or not. In the second setting, detailed in Section 5.3.3, the filter function used for the Mapper is not available and an estimation of this filter function has to be computed from the data. This second setting is referred to as the *estimated filter setting* in the following. It corresponds to PCA or Laplacian eigenfunctions, distance functions (such as the DTM), or regression and density estimators.

Risk of the Mapper. We study, in various settings, the problem of inferring a Reeb graph using Mappers and we use the metric d_{b} to assess the performance of the Mapper, seen as an estimator of the Reeb graph:

$$\mathbb{E}\left[d_{\mathrm{b}}\left(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})\right)\right],$$

where M_n is computed with either the exact filter f or with the inferred filter $\hat{f}$, depending on the context.

5.3.2 Reeb graph inference with exact filter

Known generative model

We first consider the exact filter setting in the simplest situation where the parameters a and b of the generative model are known. In this setting, given Rips parameter δ, gain g and resolution r, the Mapper $\mathrm{M}_n = \mathrm{M}_{r,g,\delta}(X_n, Y_n)$ is computed with $Y_n = f(X_n)$. We now tune the triple of parameters (r, g, δ) depending on the parameters a and b. Let

$$V_n(\delta_n) = \max\{|f(x) - f(x')| \,:\, x, x' \in X_n, \|x - x'\| \leq \delta_n\}. \tag{5.11}$$

We choose for g a fixed value in $\left(\frac{1}{3}, \frac{1}{2}\right)$ and we take:

$$\delta_n = 8\left(\frac{2\log(n)}{an}\right)^{\frac{1}{b}} \quad \text{and} \quad r_n = \frac{V_n(\delta_n)^+}{g}, \tag{5.12}$$

where $V_n(\delta_n)^+$ denotes a value that is strictly larger but arbitrarily close to $V_n(\delta_n)$. We give below a general upper bound on the risk of M_n with these parameters, which depends on the regularity of the filter function and on the parameters of the generative model. We show a uniform convergence over a class of possible filter functions. This class of filters necessarily depends on the support of $\mathbb{P}$, so we define the class of filters for each probability measure in $\mathcal{P}_{a,b}$. For any $\mathbb{P} \in \mathcal{P}_{a,b}$, we let $\mathcal{F}(\mathbb{P}, \omega)$ denote the set of filter functions $f : X_{\mathbb{P}} \to \mathbb{R}$ such that f is Morse-type on $X_{\mathbb{P}}$ with $\omega_f \leq \omega$.

Proposition 5.3.3. *Let ω be a modulus of continuity such that $\omega(x)/x$ is a non-increasing function on $\mathbb{R}^+$. For n large enough, the Mapper computed with parameters (r_n, g, δ_n) as defined above satisfies:*

$$\sup_{\mathbb{P}\in\mathcal{P}_{a,b}} \mathbb{E}\left[\sup_{f\in\mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right)\right] \leq C\,\omega\left(\frac{2\cdot 8^b}{a}\frac{\log(n)}{n}\right)^{\frac{1}{b}}$$

where the constant C only depends on a, b, and on the geometric parameters of the model.

Proof. We fix some parameters $a > 0$ and $b \geq 1$. First note that Assumption (5.4) is always satisfied by definition of r_n. Next, there exists $n_0 \in \mathbb{N}$ such that for any $n \geq n_0$, Assumption (5.3) is satisfied because $\delta_n \to 0$ and $\omega(\delta_n) \to 0$ as $n \to +\infty$. Moreover, n_0 can be taken the same for all $f \in \bigcup_{\mathbb{P}\in\mathcal{P}(a,b)} \mathcal{F}(\mathbb{P}, \omega)$.

Let $\varepsilon_n = d_{\mathrm{H}}(X, X_n)$. Under the (a, b)-standard assumption, it is well known that (see for instance [49, 59]):

$$\mathbb{P}\left(\varepsilon_n \geq u\right) \leq \min\left\{1, \frac{4^b}{au^b}\mathrm{e}^{-a\left(\frac{u}{2}\right)^b n}\right\}, \forall u > 0. \tag{5.13}$$

In particular, regarding the complementary of (5.5) we have:

$$\mathbb{P}\left(\varepsilon_n > \frac{\delta_n}{4}\right) \leq \min\left\{1, \frac{2^b}{2\log(n)n}\right\}. \tag{5.14}$$

Recall that $\mathrm{diam}(X_{\mathbb{P}}) \leq L$. Let $\bar{C} = \omega(L)$ be a constant that only depends on the parameters of the model. Then, for any $\mathbb{P} \in \mathcal{P}(a,b)$, we have:

$$\sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right) \leq \bar{C}. \tag{5.15}$$

For $n \geq n_0$, we have :

$$\sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right) = \sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right) \mathbb{I}_{\varepsilon_n > \delta_n/4} + \sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right) \mathbb{I}_{\varepsilon_n \leq \delta_n/4}$$

and thus

$$\begin{aligned} \mathbb{E}\left[\sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right)\right] &\leq \bar{C}\mathbb{P}\left(\varepsilon_n > \frac{\delta_n}{4}\right) + r_n + 2\omega(\delta_n) \\ &\leq \bar{C}\min\left\{1, \frac{2^b}{2\log(n)n}\right\} + \left(\frac{1+2g}{g}\right)\omega(\delta_n) \end{aligned} \tag{5.16}$$

where we have used (5.15), Theorem 5.2.1 and the fact that $V_n(\delta_n)^+$ can be chosen less or equal to $\omega(\delta_n)$. For n large enough, the first term in (5.16) is of the order of δ_n^b, which can be upper bounded by δ_n and thus by $\omega(\delta_n)$ (up to a constant) since $\omega(\delta)/\delta$ is non-increasing. Since $\frac{1+2g}{g} < 6$ because $\frac{1}{3} < g < \frac{1}{2}$, we get that the risk is bounded by $\omega(\delta_n)$ for $n \geq n_0$ up to a constant that only depends on the parameters of the model. The same inequality is of course valid for any n by taking a larger constant, because n_0 itself only depends on the parameters of the model. □

Comments on Proposition 5.3.3. Assuming that $\omega(x)/x$ is nonincreasing is not a very strong assumption. This property is satisfied in particular when ω is concave, as in the case of concave majorant (see for instance Section 6 in [62]). As expected, we see that the rate of convergence of the Mapper to the Reeb graph directly depends on the regularity of the filter function and on the parameter b which roughly represents the intrinsic dimension of the data. For Lipschitz filter functions, the rate is similar to the one for persistence diagram inference [49], namely it corresponds to the one of support estimation for the Hausdorff metric (see for instance [59] and [75]). In the other cases where the filters only admit a concave modulus of continuity, we see that the "distortion" created by the filter function slows down the convergence of the Mapper to the Reeb graph.

A lower bound. We now give a lower bound that almost matches with the upper bound of Proposition 5.3.3. To prove it, we use *Le Cam's Lemma.* The version of Le Cam's Lemma given below is from [139]—see also [76]. Recall that the total variation distance between two distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ on a measured space $(X, \mathcal{B})$ is defined by

$$\mathrm{TV}(\mathbb{P}_0, \mathbb{P}_1) = \sup_{B \in \mathcal{B}} |\mathbb{P}_0(B) - \mathbb{P}_1(B)|.$$

Moreover, if $\mathbb{P}_0$ and $\mathbb{P}_1$ have densities p_0 and p_1 with respect to the same measure λ on X, then

$$\mathrm{TV}(\mathbb{P}_0, \mathbb{P}_1) = \int_X |p_0 - p_1| d\lambda.$$

Lemma 5.3.4 (Le Cam)**.** *Let $\mathcal{P}$ be a set of distributions. For $\mathbb{P} \in \mathcal{P}$, let $\theta(\mathbb{P})$ take values in a pseudo-metric space (X, ρ). Let $\mathbb{P}_0$ and $\mathbb{P}_1$ in $\mathcal{P}$ be any pair of distributions. Let $x_1, \ldots, x_n$ be drawn i.i.d. from some $\mathbb{P} \in \mathcal{P}$. Let $\hat{\theta} = \hat{\theta}(x_1, \ldots, x_n)$ be any estimator of $\theta(\mathbb{P})$, then*

$$\sup_{\mathbb{P} \in \mathcal{P}} \mathbb{E}_{\mathbb{P}^n}\left[\rho(\theta, \hat{\theta})\right] \geq \frac{1}{8}\rho\left(\theta(\mathbb{P}_0), \theta(\mathbb{P}_1)\right)\left[1 - \mathrm{TV}(\mathbb{P}_0, \mathbb{P}_1)\right]^{2n}.$$

Proposition 5.3.5. *Let ω be a modulus of continuity of f. Then, for any estimator $\hat{\mathrm{R}}_n$ of $\mathrm{R}_f(X_{\mathbb{P}})$, we have*

$$\sup_{\mathbb{P} \in \mathcal{P}_{a,b}} \mathbb{E}\left[\sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \hat{\mathrm{R}}_n\right)\right] \geq C\,\omega\left(\frac{1}{an}\right)^{\frac{1}{b}},$$

where the constant C only depends on a, b, and on the geometric parameters of the model.

Proof. The proof follows closely Section B.2 in [48]. Let $X_0 = [0, a^{-1/b}] \subset \mathbb{R}^D$. Obviously, X_0 is a compact submanifold of $\mathbb{R}^D$. Let $\mathcal{U}(X_0)$ be the uniform measure on X_0. Let $\mathcal{P}_{a,b,X_0}$ denote the set of (a,b)-standard measures whose support is included in X_0. Let $x_0 = 0 \in X_0$ and $\{x_n\}_{n\in\mathbb{N}^*} \in X_0^{\mathbb{N}}$ such that $\|x_n - x_0\| = (an)^{-1/b}$. Now, let

$$f_0 : \begin{cases} X_0 & \to \mathbb{R} \\ x & \mapsto \omega(\|x - x_0\|) \end{cases}$$

By definition, we have $f_0 \in \mathcal{F}(\mathcal{U}(X_0), \omega)$ because $\mathrm{ExDg}(f_0) = \mathrm{Ext}_0^+(f_0) = \{(0, \omega(a^{-1/b}))\}$ since f_0 is increasing by definition of ω. Finally, given any measure $\mathbb{P} \in \mathcal{P}_{a,b,X_0}$, we let $\theta_0(\mathbb{P}) = \mathrm{R}_{f_0|_{X_\mathbb{P}}}(X_\mathbb{P})$. Then, we have:

$$\begin{aligned}
&\sup_{\mathbb{P}\in\mathcal{P}_{a,b}} \mathbb{E}\left[\sup_{f\in\mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_\mathbb{P}), \hat{\mathrm{R}}_n\right)\right] \\
&\qquad \geq \sup_{\mathbb{P}\in\mathcal{P}_{a,b,X_0}} \mathbb{E}\left[\sup_{f\in\mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_\mathbb{P}), \hat{\mathrm{R}}_n\right)\right] \\
&\qquad \geq \sup_{\mathbb{P}\in\mathcal{P}_{a,b,X_0}} \mathbb{E}\left[d_{\mathrm{b}}\left(\mathrm{R}_{f_0|_{X_\mathbb{P}}}(X_\mathbb{P}), \hat{\mathrm{R}}_n\right)\right] = \sup_{\mathbb{P}\in\mathcal{P}_{a,b,X_0}} \mathbb{E}\left[\rho\left(\theta_0(\mathbb{P}), \hat{\mathrm{R}}_n\right)\right],
\end{aligned}$$

where $\rho = d_{\mathrm{b}}$. For any $n \in \mathbb{N}^*$, we let $\mathbb{P}_{0,n} = \delta_{x_0}$ be the Dirac measure on x_0 and $\mathbb{P}_{1,n} = (1 - \frac{1}{n})\mathbb{P}_{0,n} + \frac{1}{n}\mathcal{U}([x_0, x_n])$. As a Dirac measure, $\mathbb{P}_{0,n}$ is obviously in $\mathcal{P}_{a,b,X_0}$. We now check that $\mathbb{P}_{1,n} \in \mathcal{P}_{a,b,X_0}$. Let $B(x,r)$ denote the Euclidean ball centered on x with radius $r > 0$.

- Let us study $\mathbb{P}_{1,n}(B(x_0, r))$.
  Assume $r \leq (an)^{-1/b}$. Then

  $$\mathbb{P}_{1,n}(B(x_0,r)) = 1 - \frac{1}{n} + \frac{1}{n}\frac{r}{(an)^{-1/b}} \geq \left(1 - \frac{1}{n} + \frac{1}{n}\right)\left(\frac{r}{(an)^{-1/b}}\right)^b \geq \left(\frac{1}{2} + \frac{1}{n}\right) anr^b \geq ar^b.$$

  Assume $r > (an)^{-1/b}$. Then

  $$\mathbb{P}_{1,n}(B(x_0,r)) = 1 \geq \min\{ar^b\}.$$

- Let us study $\mathbb{P}_{1,n}(B(x_n, r))$. Assume $r \leq (an)^{-1/b}$. Then

  $$\mathbb{P}_{1,n}(B(x_n,r)) = \frac{1}{n}\frac{r}{(an)^{-1/b}} \geq \frac{1}{n}\left(\frac{r}{(an)^{-1/b}}\right)^b = ar^b.$$

  Assume $r > (an)^{-1/b}$. Then

  $$\mathbb{P}_{1,n}(B(x_n,r)) = 1 \geq \min\{ar^b\}.$$

- Let us study $\mathbb{P}_{1,n}(B(x,r))$, where $x \in (x_0, x_n)$. Assume $r \leq x$. Then

  $$\mathbb{P}_{1,n}(B(x,r)) \geq \frac{1}{n}\frac{r}{(ab)^{-1/b}} \geq ar^b \text{ (see previous case).}$$

  Assume $r > x$. Then $\mathbb{P}_{1,n}(B(x,r)) = 1 - \frac{1}{n} + \frac{1}{n}\frac{\left(x + \min\{r,\ (an)^{-1/b} - x\}\right)}{(an)^{-1/b}}$. If $\min\{r,\ (an)^{-1/b} - x\} = r$, then we have

  $$\mathbb{P}_{1,n}(B(x,r)) \geq 1 - \frac{1}{n} + \frac{1}{n}\frac{r}{(ab)^{-1/b}} \geq ar^b \text{ (see previous case).}$$

  Otherwise, we have

  $$\mathbb{P}_{1,n}(B(x,r)) = 1 \geq \min\{ar^b\}.$$

Thus $\mathbb{P}_{1,n}$ is in $\mathcal{P}_{a,b,X_0}$ as well. Hence, we apply Lemma 5.3.4 to get:

$$\sup_{\mathbb{P}\in\mathcal{P}_{a,b,X_0}} \mathbb{E}\left[\rho\left(\theta_0(\mathbb{P}),\hat{\mathrm{R}}_n\right)\right] \geq \frac{1}{8}\rho(\theta_0(\mathbb{P}_{0,n}),\theta_0(\mathbb{P}_{1,n}))\left[1-\mathrm{TV}(\mathbb{P}_{0,n},\mathbb{P}_{1,n})\right]^{2n}.$$

By definition, we have:

$$\rho(\theta_0(\mathbb{P}_{0,n}),\theta_0(\mathbb{P}_{1,n})) = d_{\mathrm{b}}\left(\mathrm{R}_{f_0|_{\{x_0\}}}(\{x_0\}),\mathrm{R}_{f_0|_{[x_0,x_n]}}(\mathcal{U}[x_0,x_n])\right).$$

Since $\mathrm{ExDg}\left(\mathrm{R}_{f_0|_{\{x_0\}}}(\{x_0\})\right) = \{(0,0)\}$ and $\mathrm{ExDg}\left(\mathrm{R}_{f_0|_{[x_0,x_n]}}(\mathcal{U}[x_0,x_n])\right) = \{(f(x_0),f(x_n))\}$ because f_0 is increasing by definition of ω, it follows that

$$\rho(\theta_0(\mathbb{P}_{0,n}),\theta_0(\mathbb{P}_{1,n})) = \frac{1}{2}|f(x_n)-f(x_0)| = \frac{1}{2}\omega\left((an)^{-1/b}\right).$$

It remains to compute $\mathrm{TV}(\mathbb{P}_{0,n},\mathbb{P}_{1,n}) = \left|1-\left(1-\frac{1}{n}\right)\right| + \frac{1}{n}(an)^{-1/b} = \frac{1}{n} + o\left(\frac{1}{n}\right)$. The proposition follows then from the fact that $\left[1-\mathrm{TV}(\mathbb{P}_{0,n},\mathbb{P}_{1,n})\right]^{2n} \to \mathrm{e}^{-2}$. □

Minimax Optimality. Propositions 5.3.3 and 5.3.5 together show that, with the choice of parameters given before, M_n is minimax optimal up to a logarithmic factor $\log(n)$ inside the modulus of continuity. Note that the lower bound is also valid whether or not the coefficients a and b and the filter function f and its modulus of continuity are given.

Unknown generative model

We still assume that the exact values $Y_n = f(X_n)$ of the filter on the point could can be computed and that at least an upper bound on the modulus of continuity of the filter is known. However, the parameters a and b are not assumed to be known anymore. We adapt a subsampling approach proposed by [71]. As before, for given Rips parameter δ, gain g and resolution r, the Mapper $\mathrm{M}_n = \mathrm{M}_{r,g,\delta}(X_n,Y_n)$ is computed with $Y_n = f(X_n)$.

We introduce the sequence $s_n = \frac{n}{(\log n)^{1+\beta}}$ for some fixed value $\beta > 0$. Let $X_n^{s_n}$ be an arbitrary subset of X_n that contains s_n points. We tune the triple of parameters (r,g,δ) as follows: we choose for g a fixed value in $\left(\frac{1}{3},\frac{1}{2}\right)$ and we take:

$$\delta_n = d_{\mathrm{H}}(X_n^{s_n},X_n) \quad \text{and} \quad r_n = \frac{V_n(\delta_n)^+}{g}, \tag{5.17}$$

where V_n is defined as in Equation (5.11) and d_{H} denotes the Hausdorff distance in the Euclidean norm.

Proposition 5.3.6. *Let ω be a modulus of continuity such that $x \mapsto \omega(x)/x$ is a nonincreasing function. Then, using the same notations as in the previous section, the Mapper M_n computed with parameters (r_n,g,δ_n) defined before satisfies*

$$\sup_{\mathbb{P}\in\mathcal{P}_{a,b}} \mathbb{E}\left[\sup_{f\in\mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}),\mathrm{M}_n\right)\right] \leq C\,\omega\left(\frac{C'\log(n)^{2+\beta}}{n}\right)^{\frac{1}{b}},$$

where the constants C, C' only depends on a, b, and on the geometric parameters of the model.

Proof. Using the same notation as in the previous section, we have

$$\begin{aligned}\mathbb{P}\left(\delta_n \geq u\right) &\leq \mathbb{P}\left(d_{\mathrm{H}}(X_n,X_{\mathbb{P}}) \geq \frac{u}{2}\right) + \mathbb{P}\left(d_{\mathrm{H}}(X_n^{s_n},X_{\mathbb{P}}) \geq \frac{u}{2}\right)\\ &\leq \mathbb{P}\left(\varepsilon_n \geq \frac{u}{2}\right) + \mathbb{P}\left(\varepsilon_{s_n} \geq \frac{u}{2}\right).\end{aligned} \tag{5.18}$$

Note that for any $f \in \mathcal{F}(\mathbb{P},\omega)$, according to (5.6) and (5.15)

$$d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}),\mathrm{M}_n\right) \leq (r+2\omega(\delta))\,\mathbb{I}_{\Omega_n} + \bar{C}\,\mathbb{I}_{\Omega_n^c} \tag{5.19}$$

where Ω_n is the event defined by

$$\Omega_n = \{4\delta_n \leq \min\{\kappa, \rho\}\} \cap \{4\varepsilon_n \leq \delta_n\}.$$

This gives

$$\mathbb{E}\left[\sup_{f\in\mathcal{F}(P,\omega)} d_{\mathrm{b}}\left(\mathrm{M}_n, \mathrm{R}_f(X)\right)\right] \leq \underbrace{\int_0^{\bar{C}} \mathbb{P}\left(\omega(\delta_n) \geq \frac{g}{1+2g}\alpha\right) \mathrm{d}\alpha}_{(A)} + \underbrace{\bar{C}\mathbb{P}\left(\varepsilon_n \geq \frac{\delta_n}{4}\right)}_{(B)}$$
$$+ \underbrace{\bar{C}\mathbb{P}\left(\delta_n \geq \min\left\{\frac{\kappa}{4}, \frac{\rho}{4}\right\}\right)}_{(C)}.$$

Let us bound the three terms (A), (B) and (C).

- **Term** (C). It can be bounded using (5.18) then (5.13).
- **Term** (B). Let $t_n = 2\left(\frac{2\log(n)}{an}\right)^{1/b}$ and $A_n = \{\varepsilon_n < t_n\}$. We first prove that $\delta_n \geq 4\varepsilon_n$ on the event A_n, for n large enough. We follow the lines of the proof of Theorem 3 in Section 6 in [71]. Let q_n be the t_n-*packing number* of $X_{\mathbb{P}}$, i.e. the maximal number of Euclidean balls $B(x, t_n) \cap X_{\mathbb{P}}$, where $x \in X_{\mathbb{P}}$, that can be packed into $X_{\mathbb{P}}$ without overlap. It is known (see for instance Lemma 17 in [71]) that $q_n = \Theta(t_n^{-d})$, where d is the (intrinsic) dimension of $X_{\mathbb{P}}$. Let $\mathrm{Pack}_n = \{c_1, \cdots, c_{q_n}\}$ be a corresponding packing set, i.e. the set of centers of a family of balls of radius t_n whose cardinality achieves the packing number q_n. Note that $d_{\mathrm{H}}(\mathrm{Pack}_n, X_{\mathbb{P}}) \leq 2t_n$. Indeed, for any $x \in X_{\mathbb{P}}$, there must exist $c \in \mathrm{Pack}_n$ such that $\|x - c\| \leq 2t_n$, otherwise x could be added to Pack_n, contradicting the fact that Pack_n is maximal. By contradiction, assume $\epsilon_n < t_n$ and $\delta_n \leq 4\epsilon_n$. Then:

$$\begin{aligned} d_{\mathrm{H}}(X_n^{s_n}, \mathrm{Pack}_n) &\leq d_{\mathrm{H}}(X_n^{s_n}, X_n) + d_{\mathrm{H}}(X_n, X_{\mathbb{P}}) + d_{\mathrm{H}}(X_{\mathbb{P}}, \mathrm{Pack}_n) \\ &\leq 5d_{\mathrm{H}}(X_n, X_{\mathbb{P}}) + 2t_n \leq 7t_n. \end{aligned}$$

  Now, one has $\frac{s_n}{q_n} = \Theta\left(\frac{n^{1-b/d}}{\log(n)^{1-b/d+\beta}}\right)$. Since $b \geq D \geq d$ by definition, it follows that $s_n = o(q_n)$. In particular, this means that $d_{\mathrm{H}}(X_n^{s_n}, \mathrm{Pack}_n) > 7t_n$ for n large enough, which yields a contradiction. Hence, one has $\delta_n \geq 4\varepsilon_n$ on the event A_n. Then:

$$\mathbb{P}\left(\varepsilon_n \geq \frac{\delta_n}{4}\right) \leq \underbrace{\mathbb{P}\left(\varepsilon_n \geq \frac{\delta_n}{4} \mid A_n\right)}_{=0} \mathbb{P}\left(A_n\right) + \mathbb{P}\left(A_n^c\right) = \mathbb{P}\left(A_n^c\right).$$

  Finally, the probability of A_n^c is bounded with (5.13):

$$\mathbb{P}\left(A_n^c\right) \leq \frac{2^b}{2\log(n)n}.$$

- **Term** (A). This is the dominating term. First, note that since ω is increasing, one has for all $u > 0$:

$$\mathbb{P}\left(\omega(\delta_n) \geq u\right) = \mathbb{P}\left(\delta_n \geq \omega^{-1}(u)\right). \tag{5.20}$$

  Then, using (5.18) and (5.20), we have:

$$(A) \leq \int_0^{\bar{C}} \mathbb{P}\left(\varepsilon_n \geq \frac{1}{2}\omega^{-1}\left(\frac{g\alpha}{1+2g}\right)\right) \mathrm{d}\alpha + \int_0^{\bar{C}} \mathbb{P}\left(\varepsilon_{s_n} \geq \frac{1}{2}\omega^{-1}\left(\frac{g\alpha}{1+2g}\right)\right) \mathrm{d}\alpha.$$

  We only bound the first integral, but the analysis extends verbatim to the second integral when replacing n by s_n. Let

$$\alpha_n = \frac{1+2g}{g}\omega\left[\left(\frac{4^b\log(n)}{an}\right)^{1/b}\right].$$

Since $x \mapsto \frac{\omega(x)}{x}$ is non-increasing, it follows that $x \mapsto \frac{\omega^{-1}(x)}{x}$ is non-decreasing, and

$$\omega^{-1}(x) \geq \frac{x}{y}\omega^{-1}(y), \ \forall x \geq y > 0. \tag{5.21}$$

Taking inspiration from Section B.2 in [48] and using (5.13), we derive the following inequalities:

$$\begin{aligned}
\int_0^{\bar{C}} \mathbb{P}\left(\varepsilon_n \geq \frac{1}{2}\omega^{-1}\left(\frac{g\alpha}{1+2g}\right)\right) \mathrm{d}\alpha &\leq \alpha_n + \frac{8^b}{a}\int_{\alpha_n}^{\bar{C}} \frac{1}{\omega^{-1}\left(\frac{g\alpha}{1+2g}\right)^b} \exp\left[-\frac{an}{4^b}\omega^{-1}\left(\frac{g\alpha}{1+2g}\right)^b\right] \mathrm{d}\alpha \\
&\leq \alpha_n + \frac{8^b}{a}\int_{\alpha_n}^{\bar{C}} \frac{\alpha_n^b}{\left[\alpha\omega^{-1}\left(\frac{g\alpha_n}{1+2g}\right)\right]^b} \exp\left[-\frac{an\alpha^b}{(4\alpha_n)^b}\omega^{-1}\left(\frac{g\alpha_n}{1+2g}\right)^b\right] \mathrm{d}\alpha \\
&\leq \alpha_n + \alpha_n \frac{2^b 4 n^{1-1/b}}{b a^{1/b}\omega^{-1}\left(\frac{g\alpha_n}{1+2g}\right)} \int_{u \geq \frac{an}{4^b}\omega^{-1}\left(\frac{g\alpha_n}{1+2g}\right)^b} u^{1/b-2}\mathrm{e}^{-u}\mathrm{d}u \\
&= \alpha_n + \alpha_n \frac{2^b n}{b\log(n)^{1/b}} \int_{u \geq \log(n)} u^{1/b-2}\mathrm{e}^{-u}\mathrm{d}u \leq \left(1 + \frac{2^b}{b\log(n)^2}\right)\alpha_n \text{ since } b \geq 1 \\
&\leq C(b)\alpha_n,
\end{aligned}$$

where we used (5.21) with $x = \frac{g\alpha}{1+2g}$ and $y = \frac{g\alpha_n}{1+2g}$ for the second inequality. The constant $C(b)$ only depends on b.

Hence, since $\frac{1+2g}{g} < 6$, there exist constants $K, K' > 0$ that depend only of the geometric parameters of the model such that:

$$(A) \leq K\omega\left(\frac{K'\log(s_n)}{s_n}\right)^{\frac{1}{b}}.$$

Final bound. Since $s_n = n\log(n)^{-(1+\beta)}$, by gathering all four terms, there exist constants $C, C' > 0$ such that:

$$\mathbb{E}\left[\sup_{f \in \mathcal{F}(\mathbb{P},\omega)} d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n\right)\right] \leq C\omega\left(\frac{C'\log(n)^{2+\beta}}{n}\right)^{\frac{1}{b}}.$$

□

Minimax Optimality. Up to logarithmic factors inside the modulus of continuity, we find that this Mapper is still minimax optimal over the class $\mathcal{P}_{a,b}$ by Proposition 5.3.5.

5.3.3 Reeb graph inference with estimated filter

In this section, we assume that the *true filter* f is unknown but can be estimated from the data using an estimator $\hat{f}$. Without loss of generality we assume that both f and $\hat{f}$ are defined over $\mathbb{R}^D$. As before, parameters a and b are not assumed to be known and we have to tune the triple of parameters (r_n, g, δ_n). In this context, the quantity V_n cannot be computed as before because there is no direct access to the values of f: we only know an estimation $\hat{f}$ of it. However, in many cases, an upper bound ω_1 on the modulus of continuity of f is known, which makes possible the tuning of the parameters. For instance, PCA (and kernel) projectors, eccentricity functions, DTM functions (see [44]) are all 1-Lipschitz functions, and Corollary 5.3.7 below can be applied.

Let

$$\hat{V}_n(\delta_n) = \max\{|\hat{f}(x) - \hat{f}(x')| \,:\, x, x' \in X_n, \|x - x'\| \leq \delta_n\}, \tag{5.22}$$

and let ω_1 be a modulus of continuity for f. Let

$$r_n = \frac{\max\{\omega_1(\delta_n), \hat{V}_n(\delta_n)\}^+}{g}. \tag{5.23}$$

Following the lines of the proof of Proposition 5.3.6 and applying Corollary 5.2.3, we obtain:

Corollary 5.3.7. *Let $f : \mathbb{R}^D \to \mathbb{R}$ be a Morse-type filter function and let $\hat{f} : \mathbb{R}^D \to \mathbb{R}$ be a Morse-type estimator of f. Let ω_1 (resp. ω_2) be a modulus of continuity for f (resp. $\hat{f}$). Let $\omega = \max\{\omega_1, \omega_2\}$ such that $x \mapsto \omega(x)/x$ is a nonincreasing function. Let also $\hat{\mathrm{M}}_n = \mathrm{M}_{r_n,g,\delta_n}(X_n, \hat{f}(X_n))$ be the Mapper built on X_n with function $\hat{f}$ and parameters g, δ_n as in Equation* (5.17)*, and r_n as in Equation* (5.23)*. Then, $\hat{\mathrm{M}}_n$ satisfies*

$$\mathbb{E}\left[d_{\mathrm{b}}\left(\mathrm{R}_f(X_{\mathbb{P}}), \hat{\mathrm{M}}_n\right)\right] \leq C\omega\left(\frac{C'\log(n)^{2+\beta}}{n}\right)^{\frac{1}{b}} + \mathbb{E}\left[\max_{1\leq i\leq n} |f(x_i) - \hat{f}(x_i)|\right],$$

where the constants C, C' only depends on a, b, and the geometric parameters of the model.

Note that ω_1 has to be known to compute $\hat{\mathrm{M}}_n$ in Corollary 5.3.7 since it appears in the definition of r_n in Equation (5.23). On the contrary, ω_2—and thus ω—are not required to tune the parameters.

PCA eigenfunctions. Assume that the measure μ has a finite second moment. Following [13], we define the covariance operator $\Gamma(\cdot) = \mathbb{E}(\langle X, \cdot\rangle X)$ and we let Π_k denote the orthogonal projection onto the space spanned by the k-th eigenvector of Γ. In practice, we consider the empirical version of the covariance operator

$$\hat{\Gamma}_n(\cdot) = \frac{1}{n}\sum_{i=1}^{n}\langle X_i, \cdot\rangle X_i$$

and the empirical projection $\hat{\Pi}_k$ onto the space spanned by the k-th eigenvector of $\hat{\Gamma}_n$. According to [13](see also [16, 128]), we have

$$\mathbb{E}\left[\|\Pi_k - \hat{\Pi}_k\|_\infty\right] = O\left(\frac{1}{\sqrt{n}}\right).$$

This, together with Corollary 5.3.7 and the fact that both Π_k and $\hat{\Pi}_k$ are 1-Lipschitz, gives that the rate of convergence of the Mapper of $\hat{\Pi}_k(X_n)$ computed with parameters δ_n and g as in Equation (5.17), and r_n as in Equation (5.23), i.e. $r_n = g^{-1}\delta_n^+$, satisfies

$$\mathbb{E}\left[d_{\mathrm{b}}\left(\mathrm{R}_{\Pi_k}(X_{\mathbb{P}}), \mathrm{M}_{r_n,g,\delta_n}(X_n, \hat{\Pi}_k(X_n))\right)\right] \lesssim \left(\frac{\log(n)^{2+\beta}}{n}\right)^{1/b} \vee \frac{1}{\sqrt{n}}.$$

Hence, the rate of convergence of the Mapper is not deteriorated by using $\hat{\Pi}_k$ instead of Π_k if the intrinsic dimension b of the support of μ is at least 2.

The distance to measure. It is well known that topological methods may fail in the presence of outliers. To address this issue, [44] introduced an alternative distance function which is robust to noise, the *distance-to-a-measure* (DTM). A similar analysis as with the PCA filter can be carried out with the DTM filter using the rates of convergence proven in [51].

5.4 Confidence sets for the signatures

In practice, computing a Mapper M_n as an estimator of $\mathrm{R}_f(X_{\mathbb{P}})$ is not sufficient: we need to know how accurate these estimations are. In this section, we explain how to get confidence sets for the Mapper.

5.4.1 Confidence sets

For $\alpha \in (0, 1)$, we look for some value $\eta_{n,\alpha}$ such that

$$\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})) \geq \eta_{n,\alpha}\right) \leq \alpha$$

or at least such that

$$\limsup_{n\to\infty} \mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})) \geq \eta_{n,\alpha}\right) \leq \alpha.$$

Let

$$\mathcal{M}_\alpha = \{\mathrm{R} \in \mathsf{Reeb} \,:\, d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}) \leq \alpha\}$$

be the closed ball of radius α in the bottleneck distance and centered at the Mapper M_n in the space of Reeb graphs Reeb. Following [71], we can visualize the signatures of the points belonging to this ball in various ways. One first option is to center a box of side length 2α at each point of the extended persistence diagram of M_n—see the right columns of Figure 5.3 and Figure 5.4 for instance. An alternative solution is to visualize the confidence set by adding a band at (vertical) distance 2α from the diagonal Δ. The points outside the band are then considered as significant topological features, see [71] for more details.

Related work. Several methods have been proposed in [71] and [46] to define confidence sets for persistence diagrams. We now adapt these ideas to provide confidence sets for Mappers. Except for the bottleneck bootstrap (see further), all the methods proposed in these two articles rely on the stability results for persistence diagrams, which say that persistence diagrams equipped with the bottleneck distance are stable under Hausdorff perturbations of the data—see Theorem 2.2.17. Confidence sets for diagrams are then directly derived from confidence sets in the sample space. Here, we follow a similar strategy using Theorem 5.2.1, as explained in the next section.

5.4.2 Confidence sets derived from Theorem 5.2.1

In this section, we always assume that an upper bound ω on the exact modulus of continuity ω_f of the filter function is known. We start with the following remark: if we can take δ of the order of $d_{\mathrm{H}}(X_{\mathbb{P}}, X_n)$ in Theorem 5.2.1 and if all the conditions of the theorem are satisfied, then $d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}}))$ can be bounded in terms of $\omega(d_{\mathrm{H}}(X_{\mathbb{P}}, X_n))$. This means that we can adapt the methods of [71] to Mappers.

Known generative model. Let us first consider the simplest situation where the parameters a an b are also known. Following Section 5.3.2, we choose for g a fixed value in $\left(\frac{1}{3}, \frac{1}{2}\right)$ and we take

$$\delta_n = 8\left(\frac{2\log(n)}{an}\right)^{1/b} \quad \text{and} \quad r_n = \frac{V_n(\delta_n)^+}{g},$$

where V_n is defined as in Equation (5.11). Let $\varepsilon_n = d_{\mathrm{H}}(X_{\mathbb{P}}, X_n)$. As shown in the proof of Proposition 5.3.3, for n large enough, Assumption (5.3) and (5.4) are always satisfied and then

$$\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})) \geq \eta\right) \leq \mathbb{P}\left(\delta_n \geq \omega^{-1}\left(\frac{\eta}{2+g^{-1}}\right)\right).$$

Consequently,

$$\begin{aligned}
\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})) \geq \eta\right) &\leq \mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})) \geq \eta \cap \varepsilon_n \leq 4\delta_n\right) + \mathbb{P}\left(\varepsilon_n > 4\delta_n\right) \\
&\leq \mathbb{I}_{\omega(\delta_n) \geq \frac{g}{1+2g}\eta} + \min\left\{1, \frac{2^b}{2\log(n)n}\right\} \\
&= \Phi_n(\eta),
\end{aligned}$$

where Φ_n depends on the parameters of the model (or some bounds on these parameters) which are here assumed to be known. Hence, given a probability level α, one has:

$$\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n, \mathrm{R}_f(X_{\mathbb{P}})) \geq \Phi_n^{-1}(\alpha)\right) \leq \alpha.$$

Unknown generative model. We now assume that a and b are unknown. To compute confidence sets for the Mapper in this context, we approximate the distribution of $d_{\mathrm{H}}(X_{\mathbb{P}}, X_n)$ using the distribution of $d_{\mathrm{H}}(\hat{X}_n^{s_n}, X_n)$ conditionally to X_n. There are $N_1 = \binom{n}{s_n}$ subsets of size s_n inside X_n, so we let $X_{s_n}^1, \ldots, X_{s_n}^{N_1}$ denote all the possible configurations. Define

$$L_n(t) = \frac{1}{N_1}\sum_{k=1}^{N_1} \mathbb{I}_{d_{\mathrm{H}}\left(X_{s_n}^k, X_n\right) > t}.$$

Let s be the function on $\mathbb{N}$ defined by $s(n) = s_n$ and let $s_n^2 = s(s(n))$. There are $N_2 = \binom{n}{s_n^2}$ subsets of size s_n^2 inside X_n. Again, we let $X_{s_n^2}^k$, $1 \leq k \leq N_2$, denote these configurations and we also introduce

$$F_n(t) = \frac{1}{N_2} \sum_{k=1}^{N_2} \mathbb{I}_{d_{\mathrm{H}}\left(X_{s_n^2}^k, X_{s_n}\right) > t}.$$

Proposition 5.4.1. *Let $\eta > 0$. Then, one has the following confidence set:*

$$\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n) \geq \eta\right) \leq F_n\left(\frac{1}{4}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) + L_n\left(\frac{1}{4}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) + o\left(\frac{s_n}{n}\right)^{1/4}.$$

Proof. We have the following bound by using (5.19) in the proof of Proposition 5.3.6:

$$\begin{aligned}
&\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n) \geq \eta\right) \\
&\leq \mathbb{P}\left(\omega(\delta_n) \geq \frac{g}{1+2g}\eta\right) + \mathbb{P}\left(\varepsilon_n \geq \frac{\delta_n}{4}\right) + \mathbb{P}\left(\delta_n \geq \min\left\{\frac{\kappa}{4}, \frac{\rho}{4}\right\}\right) \\
&\leq \mathbb{P}\left(\varepsilon_n \geq \frac{1}{2}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) + \mathbb{P}\left(\varepsilon_{s_n} \geq \frac{1}{2}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) + o\left(\frac{1}{n\log(n)}\right).
\end{aligned}$$

Following the lines of Section 6 in [71], subsampling approximations give

$$\mathbb{P}\left(\varepsilon_n \geq \frac{1}{2}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) \leq L_n\left(\frac{1}{4}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) + o\left(\frac{s_n}{n}\right)^{1/4},$$

and

$$\mathbb{P}\left(\varepsilon_{s_n} \geq \frac{1}{2}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) \leq F_n\left(\frac{1}{4}\omega^{-1}\left(\frac{g}{1+2g}\eta\right)\right) + o\left(\frac{s_n^2}{s_n}\right)^{1/4}.$$

The result follows by taking $s_n = n\log(n)^{-(1+\beta)}$. □

Both F_n and L_n can be computed in practice, or at least approximated using Monte Carlo procedures. The upper bound on $\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{R}_f(X_{\mathbb{P}}), \mathrm{M}_n) \geq \eta\right)$ then provides an asymptotic confidence region for the persistence diagram of the Mapper M_n, which can be explicitly computed in practice. See the green squares in the first row of Figure 5.3. The main drawback of this approach is that it requires to know an upper bound on the modulus of continuity ω and, more importantly, the number of observations has to be very large, which is not the case on our examples in Section 5.5.

Modulus of continuity of the filter function. As shown in Proposition 5.4.1, the modulus of continuity of the filter function is a key quantity to describe the confidence regions. Inferring the modulus of continuity of the filter from the data is a tricky problem. Fortunately, in practice, even in the estimated filter setting, the modulus of continuity of the function is known in many situations. For instance, projections such as PCA eigenfunctions and DTM functions are 1-Lipschitz.

5.4.3 Bottleneck Bootstrap

The two methods given above require an explicit upper bound on the modulus of continuity of the filter function. Moreover, these methods both rely on the approximation result Theorem 5.2.1, which often leads to conservative confidence sets. An alternative strategy is the *bottleneck bootstrap* introduced in [46], and which we now apply to our framework.

Let $\mathbb{P}_n$ be the empirical measure defined from the sample $(x_1, y_1), \ldots, (x_n, y_n)$. Let $(x_1^*, y_1^*) \ldots, (x_n^*, y_n^*)$ be an i.i.d. sample from $\mathbb{P}_n$ and let also M_n^* be the random Mapper defined from this sample. We then take for $\eta_{n,\alpha}$ the quantity $\eta_{n,\alpha}^*$ defined by

$$\mathbb{P}\left(d_{\mathrm{b}}(\mathrm{M}_n^*, \mathrm{M}_n) > \eta_{n,\alpha}^* \,|\, X_n\right) = \alpha. \tag{5.24}$$

Note that $\eta_{n,\alpha}^*$ can be easily estimated with Monte Carlo procedures. It has been shown in [46] that the bottleneck bootstrap is valid when computing the sublevel sets of a density estimator. The validity of the

bottleneck bootstrap has not been proven for the extended persistence diagram of any distance function. For Mapper, it would require to write $d_{\mathrm{b}}(\mathrm{M}_n^*, \mathrm{M}_n)$ in terms of the distance between the extrema of the filter function and the ones of the interpolation of the filter function on the Rips graph. We leave this problem open in this thesis.

Extension of the analysis. Our analysis can actually handle the MultiNerve edge-based version $\overline{\mathrm{M}}^{\triangle}_{r,g,\delta}(X_n, Y_n)$ by replacing gr by r in Assumption (5.4) and r by $\frac{r}{2}$ in Equation (5.6) of Theorem 5.2.1—see Remark 5.2.2, and changing constants accordingly in the proofs. In particular, this improves the resolution r_n in Equation (5.17) since $g^{-1}V_n(\delta_n)$ becomes $V_n(\delta_n)$. Hence, we use this edge-based version in Section 5.5, where this improvement on the resolution r_n allows us to compensate for the low number of observations in some datasets.

5.5 Numerical experiments

In this section, we provide few examples of parameter selections and confidence regions (which are union of squares in the extended persistence diagrams) obtained with bottleneck bootstrap. The interpretation of these regions is that squares that intersect the diagonal, which are drawn in pink color, represent topological features in the Mappers that may be horizontal or artifacts due to the cover, and that may not be present in the Reeb graph. We show in Figure 5.3 various Mappers (in each node of the Mappers, the left number is the cluster ID and the right number is the number of observations in that cluster) and 85 percent confidence regions computed on various datasets. All δ parameters and resolutions were computed with Equation (5.17) (the δ parameters were also averaged over $N = 100$ subsamplings with $\beta = 0.001$), and all gains were set to 40%. The code we used is available in the Gudhi C++ library [30]. The confidence regions were computed by bootstrapping data 100 times. Note that computing confidence regions with Proposition 5.4.1 is possible, but the numbers of observations in all of our datasets were too low, leading to conservative confidence regions that did not allow for interpretation. We provide examples for data with and without the presence of noise in Sections 5.5.2 and 5.5.1 respectively.

5.5.1 Mappers and confidence regions

Synthetic example. We computed the Mapper of an embedding of the Klein bottle into $\mathbb{R}^4$ with 10,000 points with the height function. In order to illustrate the conservativity of confidence regions computed with Proposition 5.4.1, we also plot these regions for an embedding with 10,000,000 points using the fact that the height function is 1-Lipschitz. Corresponding squares are drawn in green color. Their very large sizes show that Proposition 5.4.1 requires a very large number of observations in practice. See the first row of Figure 5.3.

3D shapes. We computed the Mapper of an ant shape and a human shape from [52] embedded in $\mathbb{R}^3$ (with 4,706 and 6,370 points respectively) Both Mappers were computed with the height function. One can see that the confidence squares for the features that are almost horizontal (such as the small branches in the Mapper of the ant) intersect indeed the diagonal. See the second and third rows of Figure 5.3.

Miller-Reaven dataset. The first dataset comes from the Miller-Reaven diabetes study that contains 145 observations of patients suffering or not from diabete. Observations were mapped into $\mathbb{R}^5$ by computing various medical features. Data can be obtained in the "locfit" R-package. In [118], the authors identified two groups of diseases with the projection pursuit method, and in [129], the authors applied Mapper with hand-crafted parameters to get back this result. Here, we normalized the data to zero mean and unit variance, and we obtained the two flares in the Mapper computed with the eccentricity function. Moreover, these flares are at least 85 percent sure since the confidence squares on the corresponding points in the extended persistence diagrams do not intersect the diagonal. See the first row of Figure 5.4.

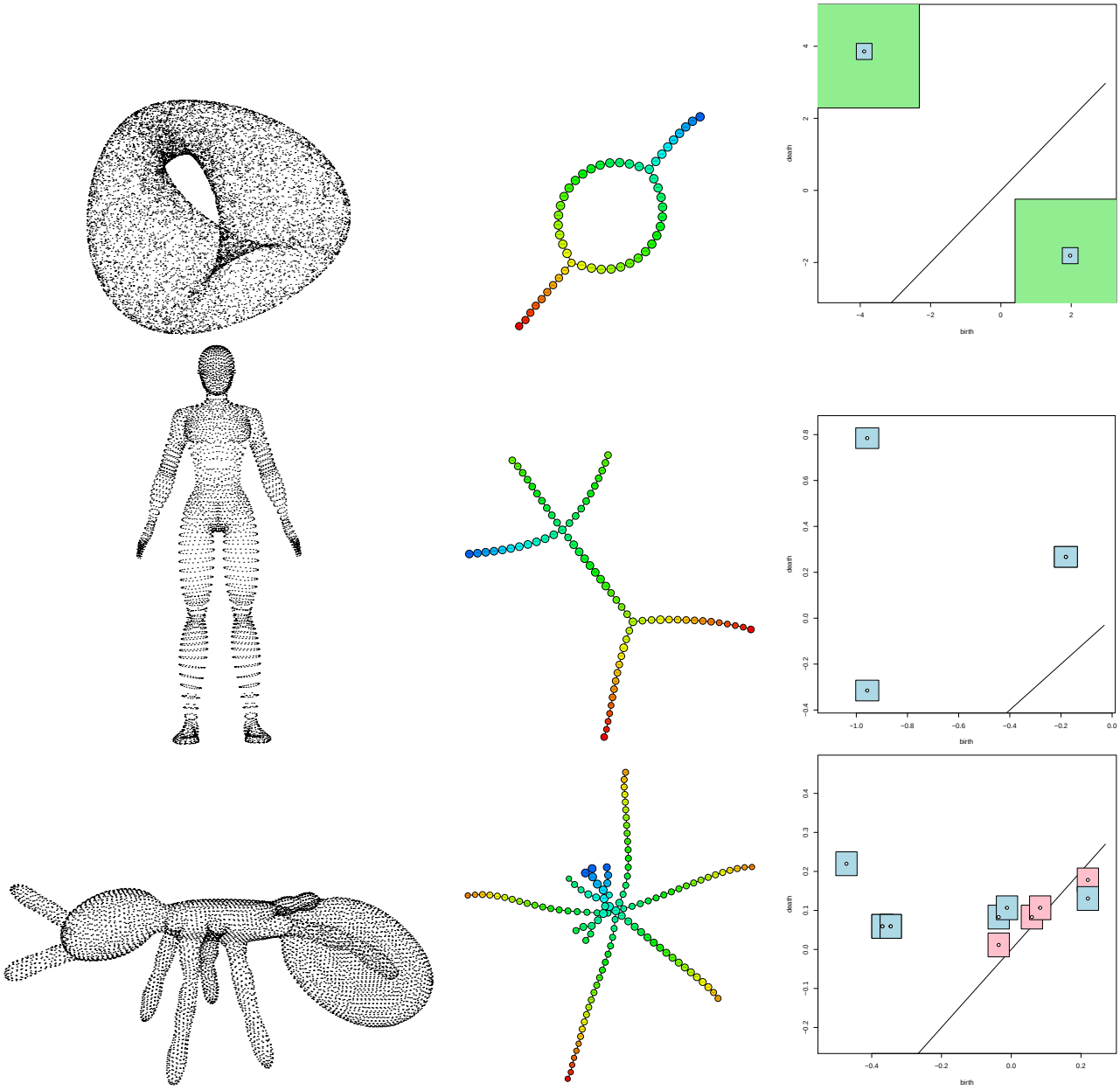

Figure 5.3: Mappers computed with automatic tuning (middle) and 85 percent confidence regions for their topological features (right) are provided for an embedding of the Klein Bottle into $\mathbb{R}^4$ (first row), a 3D human shape (second row) and a 3D ant shape (third row).

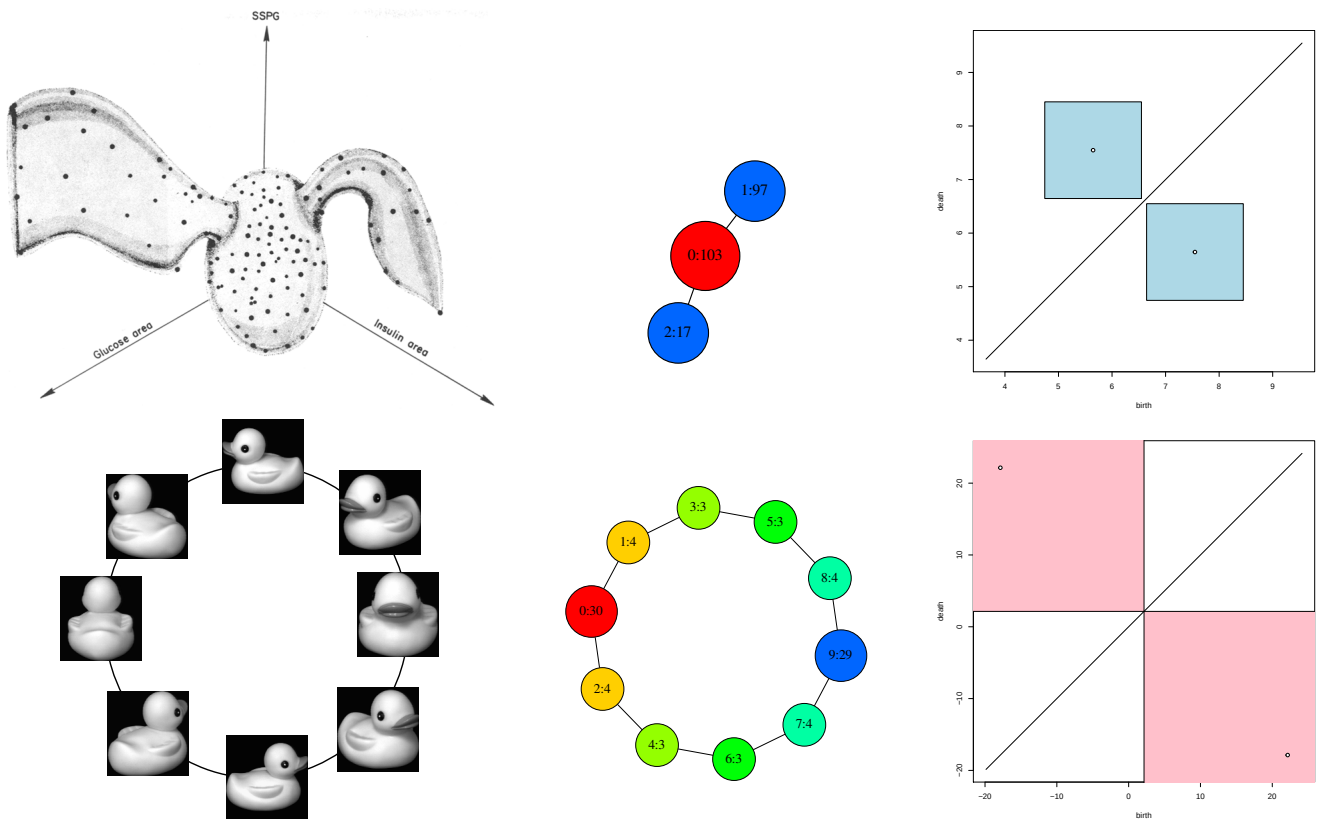

Figure 5.4: Mappers computed with automatic tuning (middle) and 85 percent confidence regions for their topological features (right) are provided for the Reaven-Miller dataset (first row) and the COIL dataset (second row).

COIL dataset. The second dataset is an instance of the 16,384-dimensional COIL dataset [107]. It contains 72 observations, each of which being a picture of a duck taken at a specific angle. Despite the low number of observations and the large number of dimensions, we managed to retrieve the intrinsic loop lying in the data using the first PCA eigenfunction. However, the low number of observations made the bootstrap fail since the confidence squares computed around the points that represent this loop in the extended persistence diagram intersect the diagonal. See the second row of Figure 5.4.

5.5.2 Noisy data

Denoising Mapper. An important drawback of the Mapper is its sensitivity to noise and outliers. See the crater dataset in Figure 5.5, for instance. Several answers have been proposed for recovering the correct persistence homology from noisy data. The idea is to use an alternative filtration of simplical compexes instead of the Rips filtration. A first option is to consider the upper level sets of a density estimator rather then the distance to the sample (see Section 4.4 in [71]). Another solution is to consider the sublevel sets of the DTM and apply persistence homology inference in [46].

Crater dataset. To handle noise in our crater dataset, we simply smoothed the dataset by computing the empirical DTM with 10 neighbors on each point and removing all points with DTM less than 40 percent of the maximum DTM in the dataset. Then we computed the Mapper with the height function. One can see that all topological features in the Mapper that are most likely artifacts due to noise (like the small loops and connected components) have corresponding confidence squares that intersect the diagonal in the extended persistence diagram. See Figure 5.5.

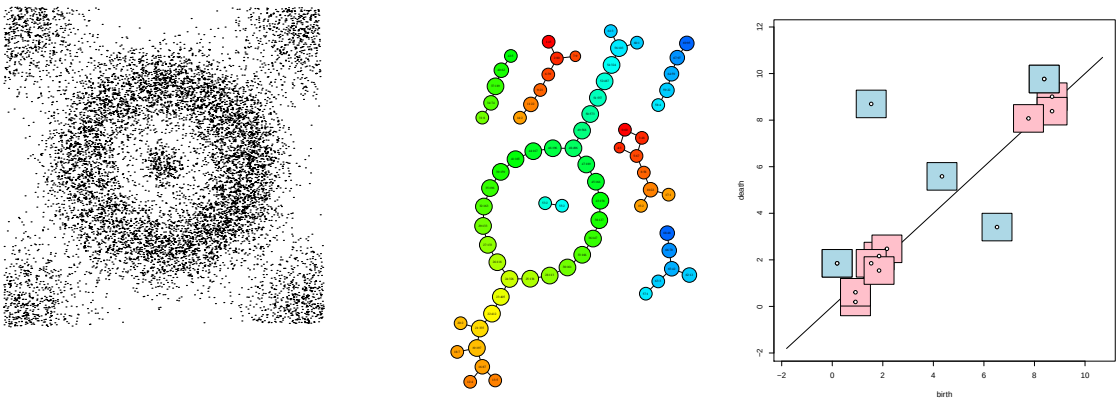

Figure 5.5: Mappers computed with automatic tuning (middle) and 85 percent confidence regions for their topological features (right) are provided for a a noisy crater in the Euclidean plane.

5.6 Conclusion

In this chapter, we studied Mappers computed on point clouds. More precisely, we derived approximation results in the deterministic case, where there is no assumptions on the point cloud generation, and we provided a statistical analysis of the Mapper when the point cloud is drawn from a probability distribution. Namely, we first proved the fact that the Mapper is a measurable construction and then we used the approximation results to show that the Mapper is a minimax optimal estimator of the Reeb graph in various contexts (Propositions 5.3.3, 5.3.5 and 5.3.6) and that corresponding confidence regions can be computed. Along the way, we derived rules of thumb to automatically tune the parameters of the Mapper with Equation (5.17), and showed their efficiency in a few examples of application of our methods on various datasets.

Among the future perspectives of this work are the following questions:

- **Can results from [46] be adapted to prove the validity of bootstrap methods?** We only used bootstrap methods empirically in this thesis. As mentioned in Section 5.4.3, proving the validity of bootstrap in the context for the Mapper would require to write $d_{\mathrm{b}}(\mathrm{M}_n^*, \mathrm{M}_n)$ in terms of the distance between the extrema of the filter function and the ones of the interpolation of the filter function on the Rips graph.
- **Is it possible to weight the Rips graph?** Using weighted Rips complexes [21] instead of the usual Rips complexes might improve the quality of the confidence regions on the Mapper features, and would probably be a better way to deal with noise than our current solution.
- **Is there applications in feature selection?** It would be interesting to check whether our statistical setting can be adapted to the question of selecting variables, which is one of the main applications of the Mapper in practice.

CHAPTER 6

KERNEL METHODS FOR PERSISTENCE DIAGRAMS

We have seen in Chapter 4 that the Mappers are stable constructions, and we presented a way in Chapter 5 to tune the parameters and build confidence sets. This is useful when the Mapper is used as a clustering method. However, Mappers can also be seen as descriptors of the data. In the context of Machine Learning, one may ask for a way to plug these descriptors in standard algorithms so as to be able to use the topological information encoded in Mappers to improve e.g. supervised learning tasks. We showed in Chapter 3 that the functional distortion distance and the bottleneck distance are locally equivalent. Hence, it makes sense to restrict the focus on the signatures, i.e. the persistence diagrams, instead of the Mappers themselves.

We recall that deriving ways to use persistence diagrams in Machine Learning is an interesting problem in its own right since their use in learning tasks is not straightforward. Indeed, a large class of learning methods, such as SVM or PCA, requires a Hilbert structure on the descriptor space, which is not the case for the space of persistence diagrams. For instance, many simple operators of $\mathbb{R}^D$, such as addition, average or scalar product, have no analogues in that space. Mapping persistence diagrams to vectors in $\mathbb{R}^D$ or in some infinite-dimensional Hilbert space is one possible approach to facilitate their use in discriminative settings, and is often referred to as *kernel methods*, such a mapping being called a kernel.

The main contribution of this chapter is to provide two ways to embed persistence diagrams into Hilbert spaces. More precisely, we define two different kernels for persistence diagrams.

The first one, called the *Sliced Wasserstein kernel* k_{SW}, is very similar to the usual Gaussian kernel, and is based on a relaxation of the 1-Wasserstein distance $d_{\mathrm{w},1}$ between persistence diagrams called the *Sliced Wasserstein distance* SW. An important result about SW is that it is *equivalent* to $d_{\mathrm{w},1}$:

$$C(N)d_{\mathrm{w},1}(\mathrm{Dg},\mathrm{Dg}') \leq \mathrm{SW}(\mathrm{Dg},\mathrm{Dg}') \leq 2\sqrt{2}d_{\mathrm{w},1}(\mathrm{Dg},\mathrm{Dg}'),$$

where $C(N)$ is a constant depending on the number of points N in Dg and Dg$'$, and such that $C(N) \to 0$ as $N \to +\infty$. We prove this result in Theorem 6.2.11.

The second one, called the *topological vector* Φ sends the persistence diagrams to a *finite* dimensional Euclidean space in a *stable* way: we show in Theorem 6.3.2 that

$$\|\Phi(\mathrm{Dg}) - \Phi(\mathrm{Dg}')\|_\infty \leq 2d_{\mathrm{b}}(\mathrm{Dg},\mathrm{Dg}').$$

Plan of the Chapter. In Section 6.1, we review the basics of supervised Machine Learning and kernel methods. We then present our Gaussian-like Sliced Wasserstein kernel in Section 6.2. Finally, we present our finite dimensional embedding in Section 6.3.

Notation. We let $\mathcal{D}_{\mathrm{f}}$ be the space of finite persistence diagrams, $\mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$ be the space of finite and bounded persistence diagrams, and $\mathcal{D}_N^{\mathrm{b}}$ be the space of bounded persistence diagrams with less than N points. We also assume to work with usual persistence diagrams, even though all definitions in this chapter hold for extended persistence diagrams by treating points type by type.

6.1 Supervised Machine Learning

In this section, we briefly recall the basics of supervised Machine Learning and kernel methods. We refer the interested reader to [72] and [127] for further details.

In the context of supervised Machine Learning, you are given n observations $(x_1, y_1), \cdots, (x_n, y_n) \in X \times Y$, where X is the space of data and Y is the space of targets—generally, targets are discrete labels in classification, and continuous variables in regression for instance. The goal is to produce a predictor $f_n : X \to Y$, which is built only from the observations: $f_n = f_n((x_1, y_1), \cdots, (x_n, y_n))$ and as accurate as possible. Accuracy is usually measured with *loss functions*, that we now detail.

6.1.1 Empirical Risk Minimization

Predictors in supervised Machine Learning are computed as the minima of the following general equation:

$$f^* = \operatorname{argmin}_{f \in \mathcal{F}} \mathrm{ER}_n(f) = \operatorname{argmin}_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f), \tag{6.1}$$

where $\mathcal{F}$ is a class of predictors, $L : Y \times Y \to \mathbb{R}$ is a *loss function* measuring the error made by f on the training observations, $\Omega(f)$ is a *regularization term* used to avoid overfitting and too complicated predictors, and $\mathrm{ER}_n(f)$ is called the *empirical risk* of f.

Loss functions. Several different loss functions exist in the literature, each corresponding to a specific Machine Learning algorithm. Assuming $Y \subseteq \mathbb{R}$, examples of such losses include:

- $L(y_i, f(x_i)) = \delta_{y_i = f(x_i)}$, known as the *zero-one loss*. Due to its non smoothness, minimizing the empirical risk with this loss may become NP-hard, even for simple classes of predictors.
- $L(y_i, f(x_i)) = \max\{0, 1 - y_i f(x_i)\}$, known as the *hinge loss*. It is used in Support Vector Machine prediction. Even though it is not smooth, the empirical risk can be minimized efficiently with it.
- $L(y_i, f(x_i)) = \log(1 + \exp(-y_i f(x_i)))$, known as the *log loss*. It is used in Logistic regression.
- $L(y_i, f(x_i)) = \exp(-y_i f(x_i))$, known as the *exponential loss*. It is used in Adaboost prediction.
- $L(y_i, f(x_i)) = (y_i - f(x_i))^2$, known as the *squared loss*. It is used in least square regression.

Regularization term. Regularization terms are often used when the class $\mathcal{F}$ is parametrized by vectors of Euclidean space $\mathcal{F} = \{f_w : w \in \mathbb{R}^D\}$. In this case, the most common regularizes are:

- $\Omega(f_w) = \langle w, w \rangle = \|w\|_2^2$, known as ℓ_2 *regularization*. It is strictly convex and differentiable, hence the empirical risk can be optimized efficiently. However, the solution w^* may be *dense*, i.e. with many nonzero coordinates.
- $\Omega(f_w) = \|w\|_1$, known as ℓ_1 *regularization*. It is convex and not differentiable at 0, but the solution w^* is in general *sparse*, i.e. with just a few nonzero coordinates.
- $\Omega(f_w) = \alpha\|w\|_1 + (1 - \alpha)\|w\|_2^2$, where $0 \leq \alpha < 1$, known as *elastic net regularization*.
- $\Omega(f_w) = \|w\|_p$, where $0 < p \leq 1$, known as ℓ_p *regularization*.

The difficulty of minimizing the empirical risk also depends a lot on the class of predictors $\mathcal{F}$. It is greatly simplified when $\mathcal{F}$ is a *reproducing kernel Hilbert space* (RKHS).

6.1.2 Reproducing Kernel Hilbert Space

RKHSs are Hilbert spaces of functions for which function evaluation at a specific point x can be computed with scalar products.

Definition 6.1.1. *A set $\mathcal{H} \subset \mathbb{R}^X$ forming a Hilbert space, with scalar product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$, is a* reproducing kernel Hilbert space *if there exists a function $k : X \times X \to \mathbb{R}$, called a* kernel, *such that:*

(i) $\{k_x : x \in X\} \subset \mathcal{H}$, *where $k_x : x \mapsto k(x, \cdot)$, and*

(ii) $f(x) = \langle f, k_x \rangle_{\mathcal{H}}$, *for any $x \in X$ and $f \in \mathcal{H}$.*

An equivalent definition is to require that the evaluation function $F_x : \mathcal{H} \to \mathbb{R}$ defined by $F_x(f) = f(x)$ is continuous for any $x \in X$.

Proposition 6.1.2. *The kernel of a RKHS is unique and, conversely, a function k can be the kernel of at most one RKHS. Hence, one can talk of* the *kernel of a RKHS.*

There is a useful characterization of kernels with *positive semi-definite* functions.

Theorem 6.1.3 (Moore-Aronszajn [4])**.** *A function $k : X \times X \to \mathbb{R}$ is a kernel if and only if it is positive semi-definite, i.e. $\sum_{i,j} a_i a_j k(x_i, x_j) \geq 0$ for any $a_1, \cdots, a_n \in \mathbb{R}$ and $x_1, \cdots, x_n \in X$.*

When $X = \mathbb{R}^D$, $D \in \mathbb{N}^*$, examples of such positive semi-definite functions include:

- the linear kernel: $k(x, y) = \langle x, y \rangle$,
- the polynomial kernel: $k(x, y) = (\alpha \langle x, y \rangle + 1)^\beta$, $\alpha, \beta \in \mathbb{R}$,
- the Gaussian kernel: $k(x, y) = \exp\left(-\frac{\|x-y\|_2^2}{2\sigma^2}\right)$, $\sigma > 0$.

Minimizing the empirical risk when $\mathcal{F}$ is a RKHS $\mathcal{H}$ turns out to be easy, even when $\mathcal{H}$ is infinite dimensional, as is the case for many kernels.

Theorem 6.1.4 (Representer Theorem [126])**.** *Let $(x_1, y_1), \cdots, (x_n, y_n) \in X \times Y$ be n observations, and let $k : X \times X \to \mathbb{R}$ be a kernel, i.e. a positive semi-definite function, with corresponding RKHS $\mathcal{H}$. Let $\Omega : \mathbb{R}_+ \to \mathbb{R}$ be a strictly monotonically increasing function, and $L : Y \times Y \to \mathbb{R}$ be an arbitrary loss function. Then, any function $f^* \in \mathcal{H}$ minimizing*

$$\frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(\|f\|_{\mathcal{H}})$$

is of the form $f^*(\cdot) = \sum_{i=1}^{n} \alpha_i k(x_i, \cdot)$, where $\alpha_1, \cdots, \alpha_n \in \mathbb{R}$.

In particular, computing f^* does not require to know the RKHS $\mathcal{H}$; only the evaluations of the kernel at the observations $k(x_i, x_j)$ are necessary.

The kernel trick. A direct consequence of the previous results is the following theorem:

Corollary 6.1.5. *For any kernel $k : X \times X \to \mathbb{R}$, there exists a essentially unique Hilbert space $\mathcal{H}$ and an embedding $\Phi : X \to \mathcal{H}$ such that:*

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}}.$$

Hence, any set X can be seen as a subset of a Hilbert space, as soon as there is a positive semi-definite function, or kernel, at hand. This is attractive since observations in this Hilbert space may be linearly separable, even if the observations themselves are not. This is known as the *kernel trick*. See Figure 6.1.

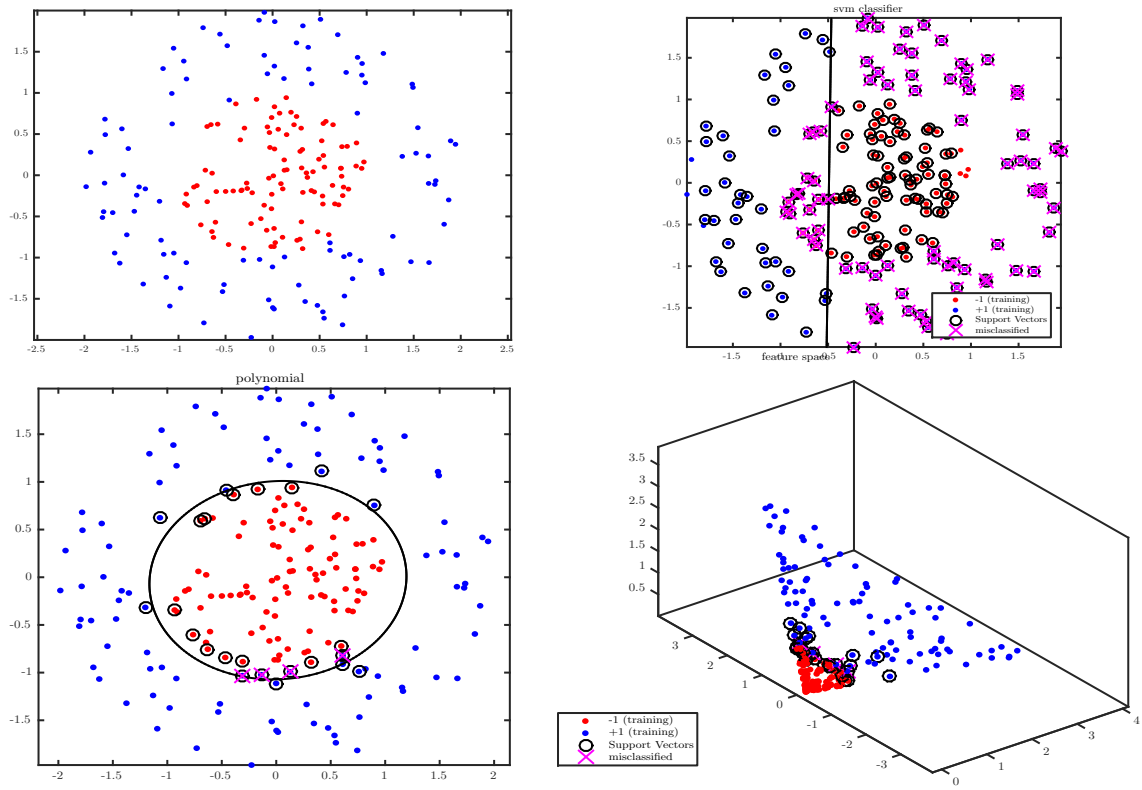

Figure 6.1: In $\mathbb{R}^2$, red points cannot be separated from blue ones with a line without producing misclassified points (first row). However, embedding these points into $\mathbb{R}^3$, for instance with a polynomial kernel, can make them separable. It then suffices to push back the separating hyperplane in $\mathbb{R}^3$ to get a non linear separation in $\mathbb{R}^2$ (second row).

Gaussian kernels. A standard way to derive a kernel is to exponentiate the negative of a squared Euclidean distance. This is due to the following result of Berg et al:

Theorem 6.1.6 (Theorem 3.2.2 of [11])**.** *Let $\sigma > 0$. The Gaussian function*

$$k_\sigma(x, y) = \exp\left(-\frac{f(x, y)}{2\sigma^2}\right),$$

for an arbitrary function f, is positive semi-definite for all $\sigma > 0$ if and only if f is a conditionally negative semi-definite *function, i.e. $\sum_{i,j} a_i a_j f(x_i, x_j) \leq 0$ for any $n \in \mathbb{N}^*$, $x_1, \cdots, x_n \in X$, and $a_1, \cdots, a_n \in \mathbb{R}$ such that $\sum_i a_i = 0$.*

Concerning persistence diagrams, it has been observed in Appendix A of [119] that, unfortunately, the metrics d_{b} or $d_{\mathrm{w},1}$ are not conditionally negative semi-definite (it suffices to randomly sample a family of point clouds to observe experimentally that more often than not the inequality of negative definiteness will be violated for particular weights $a_1, \cdots, a_n$). In the following section, we present an approximation of $d_{\mathrm{w},1}$ with the *Sliced Wasserstein distance*, which is provably conditionally negative semi-definite, and we use it to define a Gaussian kernel that can be easily tuned thanks to its bandwidth parameter σ.

6.2 A Gaussian Kernel for Persistence Diagrams

Several infinite dimensional kernels have been derived for persistence diagrams within the last few years. For instance, in [120], the authors use solutions of the heat differential equation in the plane, with initial heat sources located at the persistence diagram points, and compare them with the usual $L^2(\mathbb{R}^2)$ scalar product. Differently, in [90], the authors treat a persistence diagram as a discrete measure on the

plane, and follow by using kernel mean embeddings with Gaussian kernels—see Section 6.2.5 for precise definitions. Both kernels are provably *stable*, in the sense that the metric they induce in their respective RKHS is bounded above by the distance between persistence diagrams. Although these kernels are injective, there is no evidence that their induced RKHS distances are *discriminative*, and thus follow the geometry of the bottleneck or Wasserstein distances for persistence diagrams. In this section, we present the *Sliced Wasserstein* kernel for persistence diagrams, which is both stable and discriminative if the diagrams have bounded cardinalities. The kernel is based on a modification of the Wasserstein distance between probability measures, that we first define.

6.2.1 Wasserstein distance for unnormalized measures on $\mathbb{R}$

We first recall the basics on measures and integration. We refer the interested reader to [6] for further details.

Definition 6.2.1. *Let X be a set. A σ-*algebra *on X is a collection $\mathcal{E}$ of subsets of X such that, for any $E \in \mathcal{E}$ and countable family $\{E_n\}_{n\in\mathbb{N}}$ in $\mathcal{E}$:*

(i) $\emptyset \in \mathcal{E}$, (ii) $(X \setminus E) \in \mathcal{E}$, (iii) $\bigcup_{n\in\mathbb{N}} E_n \in \mathcal{E}$.

The pair $(X, \mathcal{E})$ is called a measurable space.

Given an arbitrary family $\mathcal{S}$ of subsets of X, the *σ-algebra generated by $\mathcal{S}$* is the smallest σ-algebra containing every element of $\mathcal{S}$. If X is a topological space, the σ-algebra generated by the open sets of X is called the *Borel algebra*.

Definition 6.2.2. *A* measure *on a measurable space $(X, \mathcal{E})$ is a function $\mu : \mathcal{E} \to \mathbb{R} \cup \{+\infty\}$ such that, for any $E \in \mathcal{E}$ and countable family of* pairwise disjoint sets *$\{E_n\}_{n\in\mathbb{N}}$ in $\mathcal{E}$:*

(i) $\mu(E) \geq 0$, (ii) $\mu(\emptyset) = 0$, (iii) $\mu\left(\bigcup_{n\in\mathbb{N}} E_n\right) = \sum_{n\in\mathbb{N}} \mu(E_n)$.

A probability *measure, sometimes called* normalized *measure, is a measure that also satisfies $\mu(E) \in [0, 1]$ for any $E \in \mathcal{E}$ and $\mu(X) = 1$.*

Definition 6.2.3. *Let $(X, \mathcal{E})$ be a measurable space. Let $f : X \to \mathbb{R}_+$ be a measurable function, i.e. $f^{-1}([t, +\infty)) \in \mathcal{E}$ for any $t \in \mathbb{R}$. Let μ be a measure on $(X, \mathcal{E})$.*

We define the integral of f in several steps:

- *If $f = \mathbf{1}_E$ where $E \in \mathcal{E}$, then $\int_X f \mathrm{d}\mu = \mu(E)$. The function f is called an* indicator function.
- *If $f = \sum_i a_i \mathbf{1}_{E_i}$, where $a_i > 0$ and $E_i \in \mathcal{E}$, then $\int_X f \mathrm{d}\mu = \sum_i a_i \int_X \mathbf{1}_{E_i} \mathrm{d}\mu = \sum_i a_i \mu(E_i)$. The function f is called* simple.
- *In general, we define the integral of f as $\int_X f \mathrm{d}\mu = \sup\{\int_X s \mathrm{d}\mu \,:\, s$ is simple and $s \leq f\}$.*

The 1-Wasserstein distance $\mathcal{W}$ [137, §6] is a distance between probability measures. For reasons that will become clear in the next section, we focus here on a variant of that distance: the 1-Wasserstein distance for nonnecessarily normalized measures on the real line [124, §2].

Definition 6.2.4. *Let μ and ν be two measures on the real line such that $\mu(\mathbb{R}) = \nu(\mathbb{R}) = r > 0$. The 1-Wasserstein distance between μ and ν is:*

$$\mathcal{W}(\mu, \nu) = \inf_{\xi \in \Pi(\mu,\nu)} \iint_{\mathbb{R}\times\mathbb{R}} |x - y| \mathrm{d}\xi(x, y), \tag{6.2}$$

where $\mathbb{R}^2$ is equipped with the Borel algebra and $\xi \in \Pi(\mu, \nu)$ is a measure on $\mathbb{R}^2$ with marginals μ and ν, i.e. $\xi(\cdot, \mathbb{R}) = \mu$ and $\xi(\mathbb{R}, \cdot) = \nu$.

This distance enjoys two good properties: it is *conditionally negative semi-definite* and *additive*. To show this, let us define the two following distances:

$$\mathcal{Q}_r(\mu,\nu) = r\int_{[0,1]} |M^{-1}(x) - N^{-1}(x)|\mathrm{d}x \tag{6.3}$$

$$\mathcal{L}(\mu,\nu) = \inf_{f\in 1\text{-Lipschitz}} \int_{\mathbb{R}} f(x)[\mu(\mathrm{d}x) - \nu(\mathrm{d}x)], \tag{6.4}$$

where M^{-1} and N^{-1} are the quantile functions of the probability measures $\frac{1}{r}\mu$ and $\frac{1}{r}\nu$ respectively, i.e. $M(x) = \frac{1}{r}\mu((-\infty,x])$ and $N(x) = \frac{1}{r}\nu((-\infty,x])$.

Proposition 6.2.5. *We have $\mathcal{W} = \mathcal{Q}_r = \mathcal{L}$. Additionally:*
(i) *$\mathcal{Q}_r$ is conditionally negative semi-definite on the space of measures of mass r;*
(ii) *for any positive measures μ,ν,γ such that $\mu(\mathbb{R}) = \nu(\mathbb{R})$, we have $\mathcal{L}(\mu+\gamma,\nu+\gamma) = \mathcal{L}(\mu,\nu)$.*

Proof. The equality between (6.2) and (6.3) is known for probability measures on the real line—see Proposition 2.17 in [124] for instance, and can be trivially generalized to unnormalized measures. The equality between (6.2) and (6.4) is due to the well known Kantorovich duality for a distance cost [137, Particular case 5.4] which can also be trivially generalized to unnormalized measures, which proves the main statement of the proposition.

The definition of Q_r shows that the Wasserstein distance is the l_1 norm of $rM^{-1} - rN^{-1}$, and is therefore conditionally negative semi-definite (as the l_1 distance between two direct representations of μ and ν as functions rM^{-1} and rN^{-1}), proving point (i). The second statement is immediate. □

We conclude this section with an important practical remark that concerns *empirical* measures.

Definition 6.2.6. *Let $(X,\mathcal{E})$ be a measurable space. A measure μ is said to be* empirical *if there exists a finite set of points $P \subset X$ such that $\mu(E) = \mathrm{card}(E\cap P)$ for any $E \in \mathcal{E}$. In that case, we write $\mu = \sum_{p\in P}\delta_p$. Each δ_p is called a* Dirac measure *on p.*

Remark 6.2.7. *For two unnormalized empirical measures on the real line $\mu = \sum_{i=1}^n \delta_{x_i}$ and $\nu = \sum_{i=1}^n \delta_{y_i}$ of same total mass, with ordered $x_1 \le \cdots \le x_n$ and $y_1 \le \cdots \le y_n$, one has:*

$$\mathcal{W}(\mu,\nu) = \sum_{i=1}^n |x_i - y_i| = \|X - Y\|_1,$$

where $X = (x_1,\cdots,x_n) \in \mathbb{R}^n$ and $Y = (y_1,\cdots,y_n) \in \mathbb{R}^n$.

6.2.2 The Sliced Wasserstein Kernel

Sliced Wasserstein distance. Any persistence diagram Dg can be seen as an empirical measure on the plane $\mu = \sum_{p\in\mathrm{Dg}}\delta_p$. Hence, $\mathcal{W}$ can be computed on persistence diagrams. Since $\mathcal{W}$ is conditionally negative semi-definite when the measures are defined on the real line (Proposition 6.2.5 and Remark 6.2.7), the idea of the Sliced Wasserstein distance of [116] is to slice the plane with lines passing through the origin, to project the measures onto these lines where $\mathcal{W}$ is computed, and to integrate the distances between the projected measures over all possible lines.

Definition 6.2.8. *Given $\theta \in \mathbb{R}^2$ with $\|\theta\|_2 = 1$, let $L(\theta)$ denote the line $\{\lambda\theta : \lambda \in \mathbb{R}\}$, and let $\pi_\theta : \mathbb{R}^2 \to L(\theta)$ be the orthogonal projection onto $L(\theta)$. Let $\mathrm{Dg}_1, \mathrm{Dg}_2$ be two persistence diagrams, and let $\mu_1^\theta = \sum_{p\in\mathrm{Dg}_1}\delta_{\pi_\theta(p)}$ and $\mu_{1\Delta}^\theta = \sum_{p\in\mathrm{Dg}_1}\delta_{\pi_\theta\circ\pi_\Delta(p)}$, and similarly for μ_2^θ, where π_Δ is the orthogonal projection onto the diagonal Δ. Then, the* Sliced Wasserstein distance *is defined as:*

$$\mathrm{SW}(\mathrm{Dg}_1,\mathrm{Dg}_2) = \frac{1}{2\pi}\int_{\mathbb{S}_1} \mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)\mathrm{d}\theta.$$

We added the projections $\mu_{1\Delta}^\theta$ and $\mu_{2\Delta}^\theta$ because Dg_1 and Dg_2 may have different number of points, Moreover, Δ counts for nothing in d_b and $d_{\mathrm{w},1}$.

Note that, by symmetry, one can restrict on the half-circle $[-\frac{\pi}{2},\frac{\pi}{2}]$ and normalize by π instead of 2π. Since $\mathcal{W}$ is conditionally negative semi-definite, we can deduce that this is also true for SW itself.

Lemma 6.2.9. SW *is conditionally negative semi-definite on* $\mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$.

Proof. Let $n \in \mathbb{N}^*$, $a_1, ..., a_n \in \mathbb{R}$ such that $\sum_i a_i = 0$ and $\mathrm{Dg}_1, ..., \mathrm{Dg}_n \in \mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$. Given $1 \leq i \leq n$, we let $\tilde{\mu}_i^\theta = \mu_i^\theta + \sum_{q \in \mathrm{Dg}_k, k \neq i} \delta_{\pi_\theta \circ \pi_\Delta(q)}$, $\tilde{\mu}_{ij\Delta}^\theta = \sum_{p \in \mathrm{Dg}_k, k \neq i,j} \delta_{\pi_\theta \circ \pi_\Delta(p)}$ and $d = \sum_i \mathrm{card}(\mathrm{Dg}_i)$. Then:

$$\begin{aligned}
&\sum_{i,j} a_i a_j \mathcal{W}(\mu_i^\theta + \mu_{j\Delta}^\theta, \mu_j^\theta + \mu_{i\Delta}^\theta) = \sum_{i,j} a_i a_j \mathcal{L}(\mu_i^\theta + \mu_{j\Delta}^\theta, \mu_j^\theta + \mu_{i\Delta}^\theta) \\
&= \sum_{i,j} a_i a_j \mathcal{L}(\mu_i^\theta + \mu_{j\Delta}^\theta + \mu_{ij\Delta}^\theta, \mu_j^\theta + \mu_{i\Delta}^\theta + \mu_{ij\Delta}^\theta) \\
&= \sum_{i,j} a_i a_j \mathcal{L}(\tilde{\mu}_i^\theta, \tilde{\mu}_j^\theta) = \sum_{i,j} a_i a_j \mathcal{Q}_d(\tilde{\mu}_i^\theta, \tilde{\mu}_j^\theta) \leq 0
\end{aligned}$$

The result follows by linearity of integration.

□

Hence, Theorem 6.1.6 allows us to define a valid kernel on $\mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$ with:

$$k_{\mathrm{SW}}(\mathrm{Dg}_1, \mathrm{Dg}_2) = \exp\left(-\frac{\mathrm{SW}(\mathrm{Dg}_1, \mathrm{Dg}_2)}{2\sigma^2}\right). \tag{6.5}$$

6.2.3 Metric Preservation

We now give the main theoretical result concerning the Sliced Wasserstein distance, which states that k_{SW}, in addition to be stable and injective, preserves the metric between persistence diagrams, which should intuitively lead to an improvement of the classification power. This has to be compared with [120] and [90], which only prove stability and injectivity. This intuition is illustrated in Section 6.2.5 and Figure 6.6, where we show an improvement of classification accuracies on several benchmark applications.

Stability. We first give an upper bound on the Sliced Wasserstein distance.

Theorem 6.2.10. SW *is stable with respect to* $d_{\mathrm{w},1}$ *on* $\mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$*, i.e. for any* $\mathrm{Dg}_1, \mathrm{Dg}_2 \in \mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$*, one has:*

$$\mathrm{SW}(\mathrm{Dg}_1, \mathrm{Dg}_2) \leq 2\sqrt{2} d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2).$$

Proof. Let $\theta \in \mathbb{R}^2$ be such that $\|\theta\|_2 = 1$. Let $\mathrm{Dg}_1, \mathrm{Dg}_2 \in \mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$, and let $\mathrm{Dg}_1^\theta = \{\pi_\theta(p) : p \in \mathrm{Dg}_1\} \cup \{\pi_\theta \circ \pi_\Delta(q) : q \in \mathrm{Dg}_2\}$ and $\mathrm{Dg}_2^\theta = \{\pi_\theta(q) : q \in \mathrm{Dg}_2\} \cup \{\pi_\theta \circ \pi_\Delta(p) : p \in \mathrm{Dg}_1\}$. Let γ^* be the one-to-one bijection between Dg_1^θ and Dg_2^θ induced by $\mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)$, and let γ be the one-to-one bijection between $\mathrm{Dg}_1 \cup \pi_\Delta(\mathrm{Dg}_2)$ and $\mathrm{Dg}_2 \cup \pi_\Delta(\mathrm{Dg}_1)$ induced by the partial bijection achieving $d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2)$. Then γ naturally induces a one-to-one matching γ_θ between Dg_1^θ and Dg_2^θ with:

$$\gamma_\theta = \{(\pi_\theta(p), \pi_\theta(q)) : (p,q) \in \gamma\} \cup \{(\pi_\theta \circ \pi_\Delta(p), \pi_\theta \circ \pi_\Delta(q)) : (p,q) \in \gamma,\ p,q \notin \mathrm{im}(\pi_\Delta)\}.$$

Now, one has the following inequalities:

$$\begin{aligned}
&\mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta) = \sum_{(x,y) \in \gamma^*} |x - y| \\
&\leq \sum_{(\pi_\theta(p), \pi_\theta(q)) \in \gamma_\theta} |\langle p, \theta \rangle - \langle q, \theta \rangle| \text{ since } \gamma_\theta \text{ is not the optimal matching between } \mathrm{Dg}_1^\theta \text{ and } \mathrm{Dg}_2^\theta \\
&\leq \sum_{(\pi_\theta(p), \pi_\theta(q)) \in \gamma_\theta} \|p - q\|_2 \text{ by the Cauchy-Schwarz inequality since } \|\theta\|_2 = 1 \\
&\leq \sqrt{2} \sum_{(\pi_\theta(p), \pi_\theta(q)) \in \gamma_\theta} \|p - q\|_\infty \text{ since } \|\cdot\|_2 \leq \sqrt{2}\|\cdot\|_\infty \\
&\leq 2\sqrt{2} \sum_{(p,q) \in \gamma} \|p - q\|_\infty \text{ since } \|\pi_\Delta(p) - \pi_\Delta(q)\|_\infty \leq \|p - q\|_\infty \\
&= 2\sqrt{2} d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2)
\end{aligned}$$

Hence, we have $\mathrm{SW}(\mathrm{Dg}_1, \mathrm{Dg}_2) \leq 2\sqrt{2} d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2)$. □

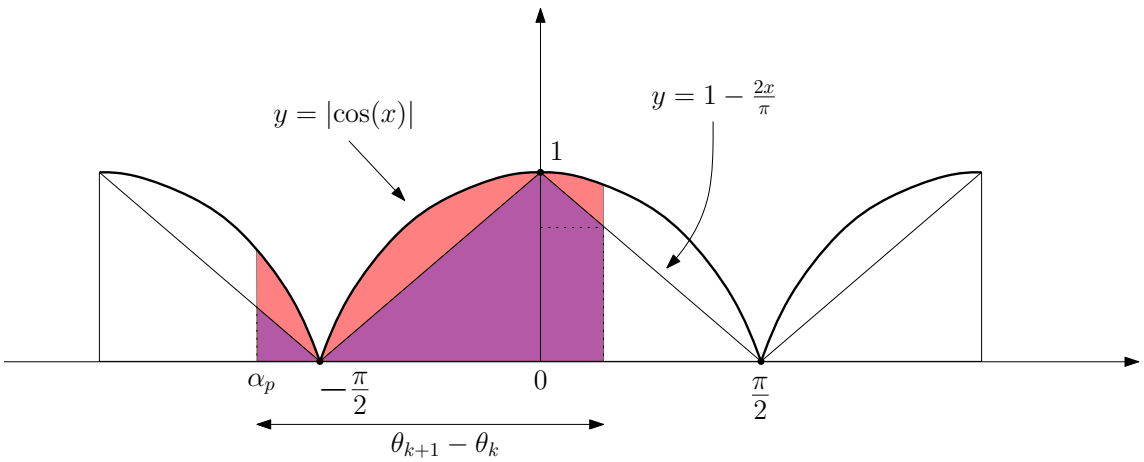

Figure 6.2: The integral of $|\cos(\cdot)|$ has a lower bound that depends on the length of the integral support. In particular, when $\theta_{k+1} - \theta_k \leq \pi$, this integral is more than $\frac{(\theta_{k+1}-\theta_k)^2}{2\pi}$ by the Cauchy-Schwarz inequality.

Discriminativity. We now prove the discriminativity of SW. For this, we need a stronger assumption on the persistence diagrams, namely that their cardinalities have to be not only finite, but also uniformly bounded by some $N \in \mathbb{N}^*$.

Theorem 6.2.11. SW *is* discriminative *with respect to* $d_{\mathrm{w},1}$ *on* $\mathcal{D}_N^{\mathrm{b}}$*, i.e. for any* $\mathrm{Dg}_1, \mathrm{Dg}_2 \in \mathcal{D}_N^{\mathrm{b}}$*, one has:*

$$\frac{1}{2M} d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2) \leq \mathrm{SW}(\mathrm{Dg}_1, \mathrm{Dg}_2),$$

where $M = 1 + 2N(2N-1)$.

Proof. Let $\mathrm{Dg}_1, \mathrm{Dg}_2 \in \mathcal{D}_N^{\mathrm{b}}$. Let $\mathbb{S}_1^+ \subseteq \mathbb{S}_1$ be the subset of the circle delimited by the angles $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. Let us consider the following set:

$$\Theta_1 = \left\{\theta \in \mathbb{S}_1^+ \,:\, \exists p_1, p_2 \in \mathrm{Dg}_1 : \langle \theta, p_2 - p_1 \rangle = 0\right\},$$

and similarly:

$$\Theta_2 = \left\{\theta \in \mathbb{S}_1^+ \,:\, \exists q_1, q_2 \in \mathrm{Dg}_2 : \langle \theta, q_2 - q_1 \rangle = 0\right\}.$$

Now, we let $\Theta = \Theta_1 \cup \Theta_2 \cup \left\{-\frac{\pi}{2}, \frac{\pi}{2}\right\}$ be the union of these sets, and sort Θ in decreasing order. One has $\mathrm{card}(\Theta) \leq 2N(2N-1) + 2 = M + 1$ since a vector θ that is orthogonal to a line defined by a specific pair of points (p_1, p_2) appears exactly once in $\mathbb{S}_1^+$.

For any θ that is between two consecutive $\theta_k, \theta_{k+1} \in \Theta$, the order of the projections onto $L(\theta)$ of the points of both Dg_1 and Dg_2 remains the same. Given any point $p \in \mathrm{Dg}_1 \cup \pi_\Delta(\mathrm{Dg}_2)$, we let $\gamma(p) \in \mathrm{Dg}_2 \cup \pi_\Delta(\mathrm{Dg}_1)$ be its matching point according to the matching given by $\mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)$. Then, one has the following equalities:

$$\begin{aligned}
&\int_{\theta_k}^{\theta_{k+1}} \mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)\, \mathrm{d}\theta \\
&\quad = \int_{\theta_k}^{\theta_{k+1}} \sum_{p \in \mathrm{Dg}_1 \cup \pi_\Delta(\mathrm{Dg}_2)} |\langle p - \gamma(p), \theta \rangle|\, \mathrm{d}\theta \\
&\quad = \sum_{p \in \mathrm{Dg}_1 \cup \pi_\Delta(\mathrm{Dg}_2)} \|p - \gamma(p)\|_2 \int_0^{\theta_{k+1}-\theta_k} |\cos(\alpha_p + \beta)|\, \mathrm{d}\beta \text{ where } \alpha_p = \angle(p - \gamma(p), \theta_k)
\end{aligned}$$

We need to lower bound $\int_0^{\theta_{k+1}-\theta_k} |\cos(\alpha_p + \beta)| d\beta$. Since $\theta_{k+1} - \theta_k \leq \pi$, one can show that this integral cannot be less than $\frac{(\theta_{k+1}-\theta_k)^2}{2\pi}$ using cosine concavity—see Figure 6.2. Hence, we now have the following lower bound:

$$\int_{\theta_k}^{\theta_{k+1}} \mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)\ \mathrm{d}\theta \geq \frac{(\theta_{k+1}-\theta_k)^2}{2\pi} \sum_{p \in \mathrm{Dg}_1 \cup \pi_\Delta(\mathrm{Dg}_2)} \|p - \gamma(p)\|_2$$
$$\geq \frac{(\theta_{k+1}-\theta_k)^2}{2\pi} \sum_{p \in \mathrm{Dg}_1 \cup \pi_\Delta(\mathrm{Dg}_2)} \|p - \gamma(p)\|_\infty \geq \frac{(\theta_{k+1}-\theta_k)^2}{2\pi} \sum_{\substack{p \notin \pi_\Delta(\mathrm{Dg}_2) \\ \text{or } \gamma(p) \notin \pi_\Delta(\mathrm{Dg}_1)}} \|p - \gamma(p)\|_\infty$$
$$\geq \frac{(\theta_{k+1}-\theta_k)^2}{2\pi} d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2).$$

Let $\Theta = \left\{\theta_1 = -\frac{\pi}{2}, \theta_2, ..., \theta_{|\Theta|} = \frac{\pi}{2}\right\}$. Then, one has:

$$\begin{aligned}
\mathrm{SW}(\mathrm{Dg}_1, \mathrm{Dg}_2) &= \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)\ \mathrm{d}\theta \\
&= \frac{1}{\pi} \sum_{k=2}^{\mathrm{card}(\Theta)} \int_{\theta_{k-1}}^{\theta_k} \mathcal{W}(\mu_1^\theta + \mu_{2\Delta}^\theta, \mu_2^\theta + \mu_{1\Delta}^\theta)\ \mathrm{d}\theta \\
&\geq \left(\sum_{k=2}^{\mathrm{card}(\Theta)} (\theta_k - \theta_{k-1})^2 \right) \frac{d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2)}{2\pi^2} \\
&\geq \frac{\pi^2}{\mathrm{card}(\Theta) - 1} \frac{d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2)}{2\pi^2} \text{ by the Cauchy-Schwarz inequality} \\
&\geq \frac{d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2)}{2M}
\end{aligned}$$

Hence, SW is discriminative. □

Theorems 6.2.10 and 6.2.11 allow us to show that $d_{k_{\mathrm{SW}}}$, the distance induced by k_{SW} in its RKHS, is also equivalent to $d_{\mathrm{w},1}$ in a broader sense: there exist continuous, positive and nondecreasing functions g, h such that $g(0) = h(0) = 0$ and $h \circ d_{\mathrm{w},1} \leq d_{k_{\mathrm{SW}}} \leq g \circ d_{\mathrm{w},1}$.

A weaker assumption. The condition on the cardinalities of the persistence diagrams can be relaxed. Indeed, one can prove that the feature map $\Phi_{k_{\mathrm{SW}}}$ induced by k_{SW} is injective when the persistence diagrams are only assumed to be finite and bounded:

Proposition 6.2.12. *The feature map $\Phi_{k_{\mathrm{SW}}}$ is continuous and injective with respect to $d_{\mathrm{w},1}$ on $\mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$.*

Proof. Note that if the persistence diagrams have bounded cardinalities, Proposition 6.2.12 is an immediate consequence of Theorem 6.2.11. One has that $\Phi_{k_{\mathrm{SW}}}$ is continous since $d_{k_{\mathrm{SW}}}$ is stable (cf Theorem 6.2.10). Now, let $\mathrm{Dg}_1, \mathrm{Dg}_2 \in \mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$. such that $d_{k_{\mathrm{SW}}}(\mathrm{Dg}_1, \mathrm{Dg}_2) = \|\Phi_{k_{\mathrm{SW}}}(\mathrm{Dg}_1) - \Phi_{k_{\mathrm{SW}}}(\mathrm{Dg}_2)\| = 0$. We necessarily have $\mathrm{SW}(\mathrm{Dg}_1, \mathrm{Dg}_2) = 0$. Assume that $d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2) > 0$. Then, there must be a point p in Dg_1 that is not in Dg_2. The Sliced Wasserstein distance being 0, there must be, for every $\theta \in \mathbb{S}_1$, a point q_θ in Dg_2 that has the same projection onto $L(\theta)$ as p: $\pi_\theta(q_\theta) = \pi_\theta(p)$, i.e. $q_\theta \in (\pi_\theta(p), p)$, the line defined by the pair $\pi_\theta(p), p$. All these lines $(\pi_\theta(p), p)$ intersect at $p \neq q_\theta$. Thus, $q_{\theta_1} \neq q_{\theta_2}$ for any $\theta_1 \neq \theta_2$, hence Dg_2 includes an infinite number of points, which is impossible since $\mathrm{Dg}_2 \in \mathcal{D}_{\mathrm{f}}^{\mathrm{b}}$. Thus, $d_{\mathrm{w},1}(\mathrm{Dg}_1, \mathrm{Dg}_2) = 0$ and $\Phi_{k_{\mathrm{SW}}}$ is injective. □

In particular, k_{SW} can be turned into a universal kernel by considering $\exp(k_{\mathrm{SW}})$ (cf Theorem 1 in [92]). This can be useful in a variety of tasks, including tests on distributions of persistence diagrams.

6.2.4 Computation

Approximate computation. In practice, k_{SW} can be approximated in $O(N\log(N))$ time using Algorithm 2. This algorithm first samples M directions in the half-circle $\mathbb{S}_1^+$; it then computes, for each sample θ_i and for each persistence diagram Dg, the scalar products between the points of Dg and θ_i, and then sorts them in a vector $V_{\theta_i}(\mathrm{Dg})$. Finally, the ℓ_1-norm between the vectors is averaged over the sampled directions: $\mathrm{SW}_M(\mathrm{Dg}_1, \mathrm{Dg}_2) = \frac{1}{M}\sum_{i=1}^{M} \|V_{\theta_i}(\mathrm{Dg}_1) - V_{\theta_i}(\mathrm{Dg}_2)\|_1$. Note that one can easily adapt the proof of Lemma 6.2.9 to show that SW_M is conditionally negative semi-definite by using the linearity of the sum. Hence, this approximation remains a kernel. If the two persistence diagrams have cardinalities bounded by N, then the running time of this procedure is $O(MN\log(N))$. This approximation of k_{SW} is useful since, as shown in Section 6.2.5, we can observe empirically that just a few directions are sufficient to get good classification accuracies.

Algorithm 2: Approximate computation of SW

Input: $\mathrm{Dg}_1 = \{p_1^1, \cdots, p_{N_1}^1\}$, $\mathrm{Dg}_2 = \{p_1^2, \cdots, p_{N_2}^2\}$, M.
Add $\pi_\Delta(\mathrm{Dg}_1)$ to Dg_2 and vice-versa.
Let $\mathrm{SW} = 0$; $\theta = -\pi/2$; $s = \pi/M$;
for $i = 1, \cdots, M$ **do**
  Store the products $\langle p_k^1, \theta\rangle$ in an array V_1;
  Store the products $\langle p_k^2, \theta\rangle$ in an array V_2;
  Sort V_1 and V_2 in ascending order;
  $\mathrm{SW} = \mathrm{SW} + s\|V_1 - V_2\|_1$;
  $\theta = \theta + s$;
end for
Output: $(1/\pi)\mathrm{SW}$;

Exact computation. A persistence diagram is said to be in *general position* if it has no triple of aligned points. If the persistence diagrams have cardinalities bounded by N, then the exact kernel computation for persistence diagrams in general position can be done in $O(N^2\log(N))$ time with Algorithm 3. In practice, given Dg_1 and Dg_2, we slightly modify them with infinitesimally small random perturbations, so that the resulting persistence diagrams $\tilde{\mathrm{Dg}}_1$ and $\tilde{\mathrm{Dg}}_2$ are in general position. We then approximate $k_{\mathrm{SW}}(\mathrm{Dg}_1, \mathrm{Dg}_2)$ arbitrarily well with $k_{\mathrm{SW}}(\tilde{\mathrm{Dg}}_1, \tilde{\mathrm{Dg}}_2)$.

6.2.5 Experiments

In this section, we compare k_{SW} to k_{PSS} and k_{PWG} on several benchmark applications for which persistence diagrams have been proven useful. We compare these kernels in terms of classification accuracies and computational cost. We review first our experimental setting, and review these tasks one by one.

Experimental setting. We implemented and used C++ code to compute kernel values in the Gudhi C++ library [31]. These values are then handled with the LIBSVM [40] implementation of C-SVM, and results are averaged over 10 runs on a 2.4GHz Intel Xeon E5530 Quad Core. The cost factor C is cross-validated in the following grid: $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. Table 6.1 summarizes the properties of the datasets we consider, namely number of labels, as well as training and test instances for each task. Figure 6.3 and 6.4 illustrate how we use persistence diagrams to represent complex data. We first describe the two baselines we considered, along with their parameterization, followed by our proposal.

PSS. The *Persistence Scale Space* kernel k_{PSS} [120] is defined as the scalar product of the two solutions of the heat diffusion equation with initial Dirac sources located at the points of the persistence diagram.

Algorithm 3: Exact computation of SW

Input: $\mathrm{Dg}_1 = \{p_1^1, \cdots, p_{N_1}^1\}$ with $|\mathrm{Dg}_1| = N_1$, $\mathrm{Dg}_2 = \{p_1^2, \cdots, p_{N_2}^2\}$ with $|\mathrm{Dg}_2| = N_2$

Let $\Theta^1 = [], \Theta^2 = [], V_1 = [], V_2 = []$, $B_1 = [[] \ldots []]$, $B_2 = [[] \ldots []]$, $\mathrm{SW} = 0$;
for $i = 1, \cdots, N_1$ **do**
  Add $p_{N_2+i}^2 = \pi_\Delta(p_i^1)$ to Dg_2;
for $i = 1, \cdots, N_2$ **do**
  Add $p_{N_1+i}^1 = \pi_\Delta(p_i^2)$ to Dg_1;
for $i = 1, 2$ **do**
  for $j = 1, \cdots, N_1 + N_2 - 1$ **do**
    for $k = j + 1, \cdots, N_1 + N_2$ **do**
      Add $\angle \left[p_j^i - p_k^i\right]^\perp \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ to Θ^i;
  Sort A^i in ascending order;
  for $j = 1, \cdots, N_1 + N_2$ **do**
    Add $\langle p_j^i, [0, -1]\rangle$ to V_i;
  Sort V_i in ascending order;
  Let $f_i : p_j^i \mapsto$ position of $\left(p_j^i, -\frac{\pi}{2}\right)$ in V_i;
  for $j = 1, \cdots, (N_1 + N_2)(N_1 + N_2 - 1)/2$ **do**
    Let k_1, k_2 such that $\Theta^i[j] = \angle \left[p_{k_1}^i - p_{k_2}^i\right]^\perp$;
    Add $\left(p_{k_1}^i, \Theta^i[j]\right)$ to $B_i\left[f_i(p_{k_1}^i)\right]$; Add $\left(p_{k_2}^i, \Theta^i[j]\right)$ to $B_i\left[f_i(p_{k_2}^i)\right]$;
    Swap $f_i(p_{k_1}^i)$ and $f_i(p_{k_2}^i)$;
  for $j = 1, \cdots, N_1 + N_2$ **do**
    Add $\left(p_j^i, \frac{\pi}{2}\right)$ to $B_i\left[f_i(p_j^i)\right]$;
for $i = 1, \cdots, N_1 + N_2$ **do**
  Let $k_1 = 0$, $k_2 = 0$;
  Let $\theta_m = -\frac{\pi}{2}$ and $\theta_M = \min\{B_1[i][k_1]_2, B_2[i][k_2]_2\}$;
  while $\theta_m \neq \frac{\pi}{2}$ **do**
    $\mathrm{SW} = \mathrm{SW} + \|B_1[i][k_1]_1 - B_2[i][k_2]_1\|_2 \int_0^{\theta_M - \theta_m} \cos(\angle\left(B_1[i][k_1]_1 - B_2[i][k_2]_1, \theta_m\right) + \theta)\mathrm{d}\theta$;
    $\theta_m = \theta_M$;
    if $\theta_M == B_1[i][k_1]_2$ **then** $k_1 = k_1 + 1$; **else** $k_2 = k_2 + 1$;
    $\theta_M = \min\{B_1[i][k_1]_2, B_2[i][k_2]_2\}$;
return $\frac{1}{\pi}\mathrm{SW}$;

Task	Training	Test	Labels
Orbit	175	75	5
Texture	240	240	24
Human	415	1618	8
Airplane	300	980	4
Ant	364	1141	5
Bird	257	832	4
FourLeg	438	1097	6
Octopus	334	1447	2
Fish	304	905	3

Table 6.1: Number of instances in the training set, the test set and number of labels.

Task	k_{PSS} (10^{-3})	k_{PWG} (1000)	k_{SW} (6)
Orbit	63.6 ± 1.2	77.7 ± 1.2	$\mathbf{83.7} \pm 0.5$
Texture	$\mathbf{98.8} \pm 0.0$	95.8 ± 0.0	96.1 ± 0.4
Task	k_{PSS}	k_{PWG}	k_{SW}
Human	68.5 ± 2.0	64.2 ± 1.2	$\mathbf{74.0} \pm 0.2$
Airplane	65.4 ± 2.4	61.3 ± 2.9	$\mathbf{72.6} \pm 0.2$
Ant	86.3 ± 1.0	87.4 ± 0.5	$\mathbf{92.3} \pm 0.2$
Bird	67.7 ± 1.8	$\mathbf{72.0} \pm 1.2$	67.0 ± 0.5
FourLeg	67.0 ± 2.5	64.0 ± 0.6	$\mathbf{73.0} \pm 0.4$
Octopus	77.6 ± 1.0	78.6 ± 1.3	$\mathbf{85.2} \pm 0.5$
Fish	76.1 ± 1.6	$\mathbf{79.8} \pm 0.5$	75.0 ± 0.4

Table 6.2: Classification accuracies (%) for the benchmark applications.

It has the following closed form expression:

$$k_{\text{PSS}}(\text{Dg}_1, \text{Dg}_2) = \frac{1}{8\pi t} \sum_{p \in \text{Dg}_1} \sum_{q \in \text{Dg}_2} \exp\left(-\frac{\|p-q\|^2}{8t}\right) - \exp\left(-\frac{\|p-\bar{q}\|^2}{8t}\right),$$

where $\bar{q} = (y, x)$ is the symmetric of $q = (x, y)$ along the diagonal. Since there is no clear heuristic on how to tune t, this parameter is chosen in the applications by ten-fold cross-validation with random 50%-50% training-test splits and with the following set of $N_{\text{PSS}} = 13$ values: 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 500 and 1000.

PWG. Let $K, p > 0$ and Dg_1 and Dg_2 be two persistence diagrams. Let k_ρ be the Gaussian kernel with parameter $\rho > 0$. Let $\mathcal{H}_\rho$ be the RKHS associated to k_ρ. Let $\mu_1 = \sum_{x \in D_1} \arctan(K\text{pers}(x)^p) k_\rho(\cdot, x) \in \mathcal{H}_\rho$ be the kernel mean embedding of Dg_1 weigthed by the diagonal distances. Let μ_2 be defined similarly. Let $\tau > 0$. The *Persistence Weighted Gaussian* kernel k_{PWG} [90, 91] is defined as the Gaussian kernel with parameter τ on $\mathcal{H}_\rho$:

$$k_{\text{PWG}}(\text{Dg}_1, \text{Dg}_2) = \exp\left(-\frac{\|\mu_1 - \mu_2\|_{\mathcal{H}_\rho}}{2\tau^2}\right).$$

The authors in [90] provide heuristics to compute K, ρ and τ and give a rule of thumb to tune p. Hence, in the applications we select p according to the rule of thumb, and we use ten-fold cross-validation with random 50%-50% training-test splits to chose K, ρ and τ. The ranges of possible values is obtained by multiplying the values computed with the heuristics with the following range of 5 factors: 0.01, 0.1, 1, 10 and 100, leading to $N_{\text{PWG}} = 5 \times 5 \times 5 = 125$ different sets of parameters.

Task	k_{PSS} (10^{-3})	k_{PWG} (1000)	k_{SW} (6)	
Orbit	$N(124 \pm 8.4)$	$N(144 \pm 14)$	$415 \pm 7.9 + NC$	
Texture	$N(165 \pm 27)$	$N(101 \pm 9.6)$	$482 \pm 68 + NC$	
Task	k_{PSS}	k_{PWG}	k_{SW}	k_{SW} (10)
Human	$N(29 \pm 0.3)$	$N(318 \pm 22)$	$2270 \pm 336 + NC$	$107 \pm 14 + NC$
Airplane	$N(0.8 \pm 0.03)$	$N(5.6 \pm 0.02)$	$44 \pm 5.4 + NC$	$10 \pm 1.6 + NC$
Ant	$N(1.7 \pm 0.01)$	$N(12 \pm 0.5)$	$92 \pm 2.8 + NC$	$16 \pm 0.4 + NC$
Bird	$N(0.5 \pm 0.01)$	$N(3.6 \pm 0.02)$	$27 \pm 1.6 + NC$	$6.6 \pm 0.8 + NC$
FourLeg	$N(10 \pm 0.07)$	$N(113 \pm 13)$	$604 \pm 25 + NC$	$52 \pm 3.2 + NC$
Octopus	$N(1.4 \pm 0.01)$	$N(11 \pm 0.8)$	$75 \pm 1.4 + NC$	$14 \pm 2.1 + NC$
Fish	$N(1.2 \pm 0.004)$	$N(9.6 \pm 0.03)$	$72 \pm 4.8 + NC$	$12 \pm 1.1 + NC$

Table 6.3: Gram matrices computation time (s) for the benchmark applications. As explained in the text, N represents the size of the set of possible parameters, and we have $N = 13$ for k_{PSS}, $N = 5 \times 5 \times 5 = 125$ for k_{PWG} and $N = 3 \times 5 = 15$ for k_{SW}. C is a constant that depends only on the training size. In all our applications, it is less than 0.1s.

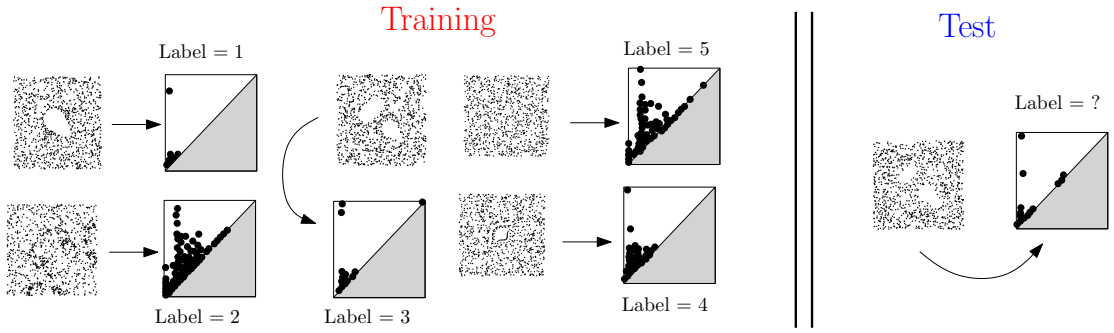

Figure 6.3: Sketch of the orbit recognition task. Each parameter r in the 5 possible choices leads to a specific behavior of the orbit. The goal is to recover parameters from the persistent homology of orbits in the test set.

Parameters for k_{SW}. The kernel we propose has only one parameter, the bandwidth σ in Eq. 6.5, which we choose using ten-fold cross-validation with random 50%-50% training-test splits. The range of possible values is obtained by computing the squareroot of the median, the first and the last deciles of all $\text{SW}(\text{Dg}_i, \text{Dg}_j)$ in the training set, then by multiplying these values by the following range of 5 factors: 0.01, 0.1, 1, 10 and 100, leading to $N_{\text{SW}} = 5 \times 3 = 15$ possible values.

Parameter Tuning. The bandwidth of k_{SW} is, in practice, easier to tune than the parameters of its two competitors when using grid search. Indeed, as is the case for all infinitely divisible kernels, the Gram matrix does not need to be recomputed for each choice of σ, since it only suffices to compute all the Sliced Wasserstein distances between persistence diagrams in the training set once. On the contrary, neither k_{PSS} nor k_{PWG} share this property, and require recomputations for each hyperparameter choice. Note however that this improvement may no longer hold if one uses other methods to tune parameters. For instance, using k_{PWG} without cross-validation is possible with the heuristics given by the authors in [90], and leads to smaller training times, but also to worse accuracies.

3D shape segmentation. Our first task is the same as in Section 6.3.3, namely we produce point classifiers for 3D shapes.

Data. We use some categories of the mesh segmentation benchmark of Chen et al. [52], which contains 3D shapes classified in several categories ("airplane", "human", "ant"...). For each category, our goal is to design a classifier that can assign, to each point in the shape, a label that describes the relative

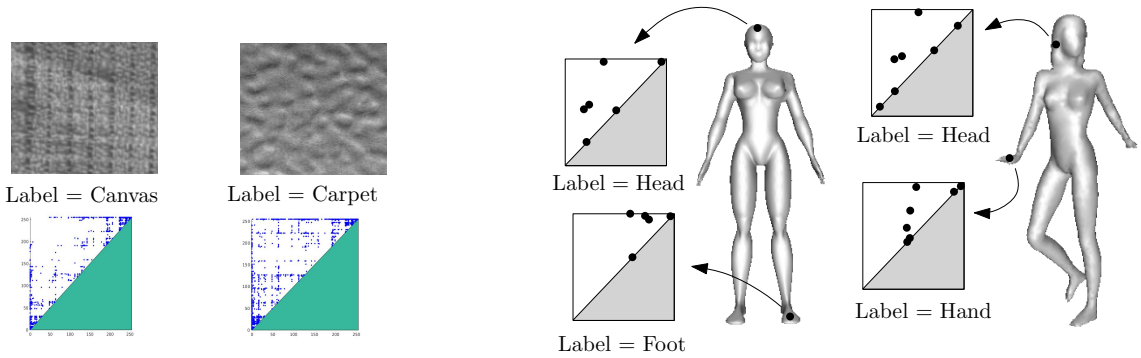

Figure 6.4: Examples of persistence diagrams computed on texture images from the *OUTEX00000* dataset and persistence diagrams computed from points on 3D shapes. One can see that corresponding points in different shapes have similar persistence diagrams.

location of that point in the shape. For instance, possible labels are, for the human category, "head", "torso", "arm"... To train classifiers, we compute a persistence diagram per point using the geodesic distance function to this point—we give more details on this persistence diagram in Section 6.3.3. For each category, the training set contains one hundredth of the points of the first five 3D shapes, and the test set contains one hundredth of the points of the remaining shapes in that category. Points in training and test sets are evenly sampled. See Figure 6.4. Here, we focus on comparison between persistence diagrams, and not on achieving state-of-the-art results. We show in Section 6.3.3 that persistence diagrams bring complementary information to classical descriptors in this task, hence reinforcing their discriminative power with appropriate kernels is of great interest. Finally, since data points are in $\mathbb{R}^3$, we set the p parameter of k_{PWG} to 5.

Results. Classification accuracies are given in Table 6.2. For most categories, k_{SW} outperforms competing kernels by a significant margin. The variance of the results over the run is also less than that of its competitors. However, training times are not better in general. Hence, we also provide the results for an approximation of k_{SW} with 10 directions. As one can see from Table 6.2 and from Figure 6.5, this approximation leaves the accuracies almost unchanged, while the training times become comparable with the ones of the other competitors. Moreover, according to Figure 6.5, using even less directions would slightly decrease the accuracies, but still outperform the competitors performances, while decreasing even more the training times.

Orbit recognition. In our second experiment, we use synthetized data. The goal is to retrieve parameters of dynamical system orbits, following an experiment proposed in [1].

Data. We study the *linked twist map*, a discrete dynamical system modeling fluid flow. It was used in [82] to model flows in DNA microarrays. Its orbits can be computed given a parameter $r > 0$ and initial positions $(x_0, y_0) \in [0,1] \times [0,1]$ as follows:

$$\begin{cases} x_{n+1} = x_n + r y_n (1 - y_n) & \mod 1 \\ y_{n+1} = y_n + r x_{n+1} (1 - x_{n+1}) & \mod 1 \end{cases}$$

Depending on the values of r, the orbits may exhibit very different behaviors. For instance, as one can see in Figure 6.3, when r is 3.5, there seems to be no interesting topological features in the orbit, while voids form for r parameters around 4.3. Following [1], we use 5 different parameters $r = 2.5, 3.5, 4, 4.1, 4.3$, that act as labels. For each parameter, we generate 100 orbits with 1000 points and random initial positions. We then compute the persistence diagrams of the distance functions to the point clouds with the Gudhi C++ library [99] and we use them (in all homological dimensions) to produce an orbit classifier that predicts the parameter values, by training over a 70%-30% training-test split of the data. Since data points are in $\mathbb{R}^2$, we set the p parameter of k_{PWG} to 4.

Results. Since the persistence diagrams contain thousands of points, we use kernel approximations to speed up the computation of the Gram matrices. In order for the approximation error to be bounded by 10^{-3}, we use an approximation of k_{SW} with 6 directions (as one can see from Figure 6.5, this has a

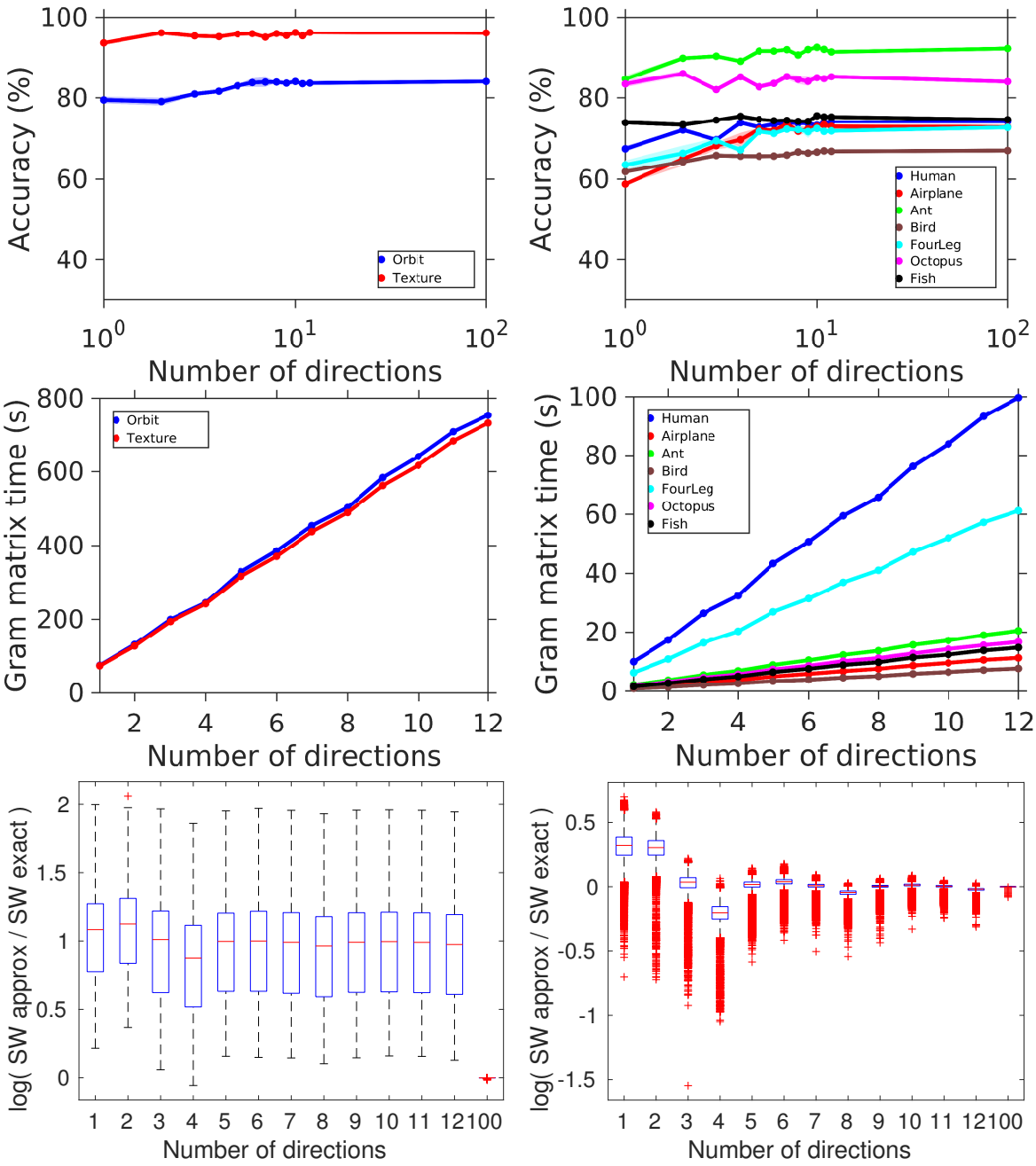

Figure 6.5: The first column corresponds to the orbit recognition and the texture classification while the second column corresponds to 3D shape segmentation. On each column, the first row shows the dependence of the accuracy on the number of directions, the second row shows the dependence of a single Gram matrix computation time, and the third row shows the dependence of the ratio of the approximation of SW and the exact SW. Since the box plot of the ratio for orbit recognition is very similar to that of 3D shape segmentation, we only give the box plot of texture classification in the first column.

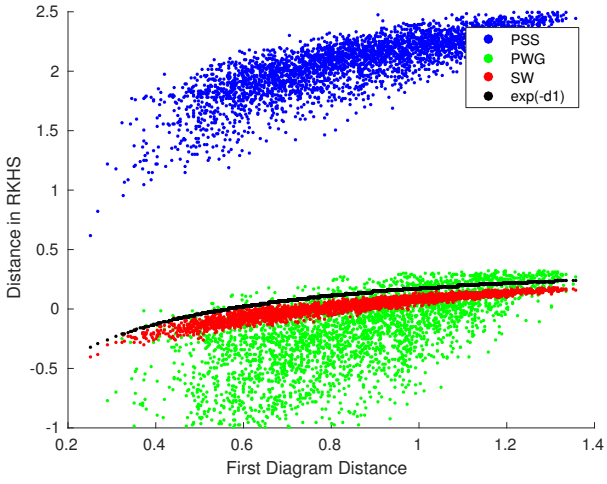

Figure 6.6: Distortion of the metric $d_{\mathrm{w},1}$. Each point represents a pair of persistence diagrams and its abscissae is the distance $d_{\mathrm{w},1}$ between them. Depending on the point color, its ordinate is the logarithm of the distance in the RKHS induced by either k_{PSS} (blue points), k_{PWG} (green points), k_{SW} (red points) and a Gaussian kernel on $d_{\mathrm{w},1}$ (black points).

small impact on the accuracy), we approximate k_{PWG} with 1000 random Fourier features [117], and we approximate k_{PSS} using Fast Gauss Transform [102] with a normalized error of 10^{-10}. One can see from Table 6.2 that the accuracy is increased a lot with k_{SW}. Concerning training times, there is also a large improvement since we tune the parameters with grid search. Indeed, each Gram matrix needs not be recomputed for each parameter when using k_{SW}.

Texture classification. Our last experiment is inspired from [120] and [94]. We use the OUTEX00000 data base [112] for texture classification.

Data. Persistence diagrams are obtained for each texture image by computing first the sign component of CLBP descriptors [79] with radius $R = 1$ and $P = 8$ neighbors for each image, and then compute the persistent homology of this descriptor using the Gudhi C++ library [67]. See Figure 6.4. Note that, contrary to the experiment of [120], we do not downsample the images to 32×32 images, but keep the original 128×128 images. Following [120], we restrict the focus to 0-dimensional persistent homology. We also use the first 50%-50% training-test split given in the database to produce classifiers. Since data points are in $\mathbb{R}^2$, we set the p parameter of k_{PWG} to 4.

Results. We use the same approximation procedure as in the previous experiment. According to Figure 6.5, even though the approximation of SW is rough, this has again a small impact on the accuracy, while reducing the training time by a significant margin. As one can see from Table 6.2, using k_{PSS} leads to almost state-of-the-art results [112, 79], closely followed by the accuracies of k_{SW} and k_{PWG}. The best timing is given by k_{SW}, again because we use grid search. Hence, k_{SW} almost achieves the best result, and its training time is better than the ones of its competitors, due to the grid search parameter tuning.

Metric Distortion. To illustrate the equivalence theorem, we also show in Figure 6.6 a scatter plot where each point represents the comparison of two persistence diagrams taken from the Airplane segmentation data set. Similar plots can be obtained with the other datasets considered here. For all points, the x-axis quantifies the 1-Wasserstein distance $d_{\mathrm{w},1}$ for that pair, while the y-axis is the logarithm of the RKHS distance induced by either k_{SW}, k_{PSS}, k_{PWG} or a Gaussian kernel directly applied to $d_{\mathrm{w},1}$, to obtain comparable quantities. We use the parameters given by the cross-validation procedure described above. One can see that the distances induced by k_{SW} are less spread than the others, suggesting that the metric induced by k_{SW} is more discriminative. Moreover the distances given by k_{SW} and the Gaussian kernel on $d_{\mathrm{w},1}$ exhibit the same behavior, suggesting that k_{SW} is the best natural equivalent of a Gaussian kernel for persistence diagrams.

6.3 Vectorization of Persistence Diagrams

We now turn the focus on finding a stable embedding into a finite dimensional Euclidean space, which may be required in a variety of tasks, such as visualization. As for infinite dimensional embeddings, a series of recent contributions have proposed vectorization methods for persistence diagrams. One can, for instance, compute and sample functions extracted from persistence diagrams [1, 20, 121], or treat the points in the persistence diagrams as roots of a complex polynomial, whose coefficients are concatenated [65].

In this section, we propose a third possibility, by sorting the entries of the distance matrices of the persistence diagrams. We first present this method and prove its stability.

6.3.1 Mapping Persistence Diagrams to Euclidean vectors

Persistence diagrams as metric spaces. To map the persistence diagrams to $\mathbb{R}^D$, we treat the diagrams themselves as finite metric spaces, and consider their distance matrices. To be oblivious to the row and column orders, we look at the distribution of the pairwise distances between the points in each diagram. For stability purposes, we also compare these pairwise distances with distance-to-diagonal terms and sort the final values. Formally:

Definition 6.3.1. *Let* $\mathrm{Dg} \in \mathcal{D}_{\mathrm{f}}$*, and let* $S = \{\min\{\|p-q\|_\infty, d_\infty(p,\Delta), d_\infty(q,\Delta)\} : p,q \in \mathrm{Dg}\}$*. The* topological map $\Phi : \mathcal{D}_{\mathrm{f}} \to \ell_\infty$ *maps* Dg *to the sequence of finite support whose first* $\mathrm{card}(S)$ *values are the elements of* S *sorted by decreasing order. If there is only one point in* Dg*, then we arbitrary set* $\Phi(\mathrm{Dg}) = 0_{\ell_\infty}$.

See Figure 6.7 for an illustration of Φ.

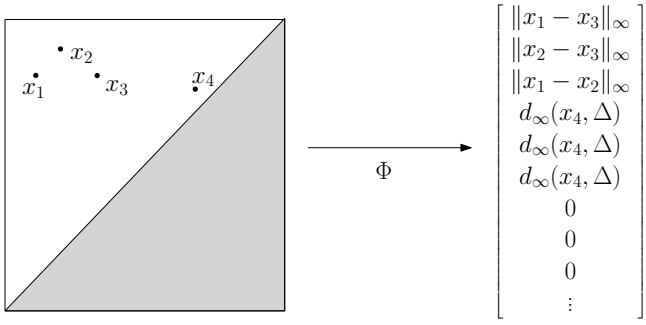

Figure 6.7: Mapping of a persistence diagram to a sequence with finite support.

Distances to the diagonal. Another solution is to keep only the sorted distances to the diagonal. Indeed, this also leads to a stable vectorization that has a significant meaning since points in persistence diagrams represent topological features—see Section 2.4.1 and 6.3.3. However, this vector alone lacks discriminative power as shown in Figure 6.8. Hence, we concatenate the two vectors in practice.

Truncation. In practice, we want to deal with finite-dimensional vectors of prescribed lengths, so we have to truncate the sequences. Since the size of the support of $\Phi(\mathrm{Dg})$ can be quadratic in the number of points in Dg, we often get rid of the last nonzero values, which are also the lowest ones. Note that this is equivalent to getting rid of pairwise terms which include either a point very close to the diagonal or two points which are very close to each other. Thus, by truncation, we either get rid of topological noise or get rid of too small distances. In the second case, it does not mean that we do not consider anymore the two points as only their mutual distance is removed while their distances to the other points are kept. In practice, we truncate the sequences according to some estimated upper bound on the number of relevant topological features in the dataset—see Section 6.3.3 for instance.

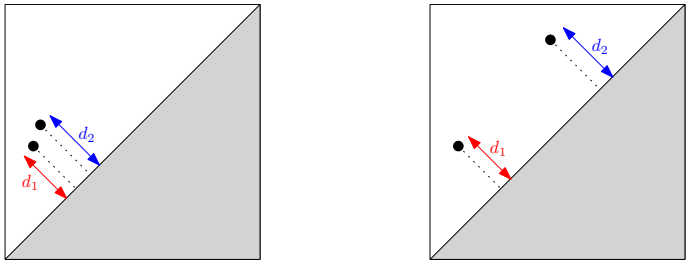

Figure 6.8: Clearly, keeping only the sorted distances to the diagonal would not discriminate the two persistence diagrams whereas looking at the distribution of the distances would allow to successfully distinguish them.

6.3.2 Stability of the topological vectors.

In this section, we prove the following stability result:

Theorem 6.3.2. *Let* $\mathrm{Dg}_1, \mathrm{Dg}_2 \in \mathcal{D}_\mathrm{f}$ *be two finite persistence diagrams. Let* $N_1 = \mathrm{card}(\mathrm{Dg}_1) > 0$, $N_2 = \mathrm{card}(\mathrm{Dg}_2) > 0$ *and* $N = \max\{N_1, N_2\}$. *Let* $D = \frac{N(N-1)}{2}$. *Then:*

$$C(N)\|\Phi(\mathrm{Dg}_1) - \Phi(\mathrm{Dg}_2)\|_2 \leq \|\Phi(\mathrm{Dg}_1) - \Phi(\mathrm{Dg}_2)\|_\infty \leq 2d_\mathrm{b}(\mathrm{Dg}_1, \mathrm{Dg}_2),$$

where $C(N) = D^{-\frac{1}{2}}$ *and* $\Phi(\mathrm{Dg}_1), \Phi(\mathrm{Dg}_2) \in \mathbb{R}^D$.

Proof. Let $\varepsilon = d_\mathrm{b}(\mathrm{Dg}_1, \mathrm{Dg}_2)$. As the problem is symmetric in Dg_1 and Dg_2, assume without loss of generality that $N_1 < N_2$. We consider one of the matchings γ^* realizing the bottleneck distance between Dg_1 and Dg_2—such matchings exist since $N_1, N_2 < +\infty$. We also call $N_{1,\gamma}$ and $N_{1,\Delta}$ the number of points of Dg_1 which are mapped by γ^* to an element of Dg_2 and to the diagonal respectively. We have $N_{1,\gamma} + N_{1,\Delta} = N_1$. Moreover, $N_{1,\gamma}$ points of Dg_2 are mapped to points of Dg_1, $N_{1,\Delta}$ points are mapped to the diagonal, and the $N_2 - N_1$ other points of Dg_2 are also mapped to the diagonal. We introduce a bijective mapping $\psi : \mathrm{Dg}_1 \to \mathrm{Dg}_2$ which coincides with γ^* on the $N_{1,\gamma}$ points of Dg_1 which are not mapped to the diagonal and which arbitrarily associates the remaining $N_{1,\Delta}$ elements of Dg_1 to $N_{1,\Delta}$ remaining points of Dg_2.

Let $V_1 = \Phi(\mathrm{Dg}_1)$ and $V_2 = \Phi(\mathrm{Dg}_2)$. By definition, we have $V_1[i] \geq V_1[i+1]$, $\forall 1 \leq i \leq N_1(N_1-1)/2$ and $V_1[i] = 0$, $\forall i > N_1(N_1-1)/2$, where $V_1[i]$ denotes the ith coordinate of V_1. Now, let $\hat{V}_2$ be the sorted vector of all $\min\{\|\psi(p_i) - \psi(p_j)\|_\infty, d_\infty(\psi(p_i), \Delta), d_\infty(\psi(p_j), \Delta)\}$, where $1 \leq i, j \leq N_1$. We also add the remaining pairwise terms of Dg_2 in $\hat{V}_2$ so that $\hat{V}_2$ has dimension $N_2(N_2-1)/2$.

We now show that $\|V_1 - \hat{V}_2\|_\infty \leq 2\varepsilon$. Fix a coordinate i. Either $i > N_1(N_1-1)/2$, and then $V_1[i] = 0$ and $\hat{V}_2[i] = \min\{\|y_{i,1} - y_{i,2}\|_\infty, d_\infty(y_{i,1}, \Delta), d_\infty(y_{i,2}, \Delta)\}$, for some $y_{i,1}, y_{i,2} \in \mathrm{Dg}_2$, or $i \leq N_1(N_1-1)/2$, and then $V_1[i] = \min\{\|x_{i,1} - x_{i,2}\|_\infty, d_\infty(x_{i,1}, \Delta), d_\infty(x_{i,2}, \Delta)\}$, and $\hat{V}_2[i] = \min\{\|\psi(x_{i,1}) - \psi(x_{i,2})\|_\infty, d_\infty(\psi(x_{i,1}), \Delta), d_\infty(\psi(x_{i,2}), \Delta)\}$, for some $x_{i,1}, x_{i,2} \in \mathrm{Dg}_1$. We have three different cases to treat here:

- (a) $i \leq \frac{N_1(N_1-1)}{2}$ and the two pairs of points are matched by γ^*,
- (b) $i \leq \frac{N_1(N_1-1)}{2}$ and at least one point of each pair is matched to Δ,
- (c) $i > \frac{N_1(N_1-1)}{2}$, and then $V_1[i] = 0$.

Case (c). In this case, at least one of the points of the pairwise term in $\hat{V}_2[i]$, say $y_{i,1}$, is matched to the diagonal. Thus, we have

$$|V_1[i] - \hat{V}_2[i]| = |\hat{V}_2[i]| \leq |d_\infty(y_{i,1}, \Delta)| \leq \varepsilon \leq 2\varepsilon.$$

Case (b). In this case, at least one point of the pairwise term in $V_1[i]$, say $x_{i,1}$, and one of the pairwise term in $\hat{V}_2[i]$, say $y_{i,1}$, are mapped to the diagonal, the other two terms being either mapped

together or to the diagonal. Then

$$|V_1[i] - \hat{V}_2[i]| \leq |d_\infty(x_{i,1}, \Delta)| + |d_\infty(y_{i,1}, \Delta)| \leq 2\varepsilon.$$

Case (a). In this case, we have $\gamma^*(x_{i,1}) = y_{i,1}$ and $\gamma^*(x_{i,2}) = y_{i,2}$. Three different subcases come out:

- (i) The minimum is in both cases the distance between the points. Then we have

$$|V_1[i] - \hat{V}_2[i]| = |\|x_{i,1} - x_{i,2}\|_\infty - \|y_{i,1} - y_{i,2}\|_\infty| \leq 2\varepsilon.$$

- (ii) The minimum is in both cases the distance of a point to the diagonal. Then either

$$|V_1[i] - \hat{V}_2[i]| = |d_\infty(x_{i,1}, \Delta) - d_\infty(y_{i,1}, \Delta)|,$$

  in which case the bound is immediate as the points are mapped by γ^*, or

$$|V_1[i] - \hat{V}_2[i]| = |d_\infty(x_{i,1}, \Delta) - d_\infty(y_{i,2}, \Delta)|,$$

  in which case we have the following inequalities:

  - $\eta_x = d_\infty(x_{i,2}, \Delta) - d_\infty(x_{i,1}, \Delta) \geq 0$,
  - $\eta_y = d_\infty(y_{i,1}, \Delta) - d_\infty(y_{i,2}, \Delta) \geq 0$,
  - $d_\infty(y_{i,1}, \Delta) = d_\infty(x_{i,1}, \Delta) + \alpha_1$ with $|\alpha_1| \leq \varepsilon$ and
  - $d_\infty(y_{i,2}, \Delta) = d_\infty(x_{i,2}, \Delta) + \alpha_2$ with $|\alpha_2| \leq \varepsilon$.

  Thus $\varepsilon \geq \alpha_1 = \eta_x + \eta_y + \alpha_2 \geq \alpha_2 + \eta_x \geq -\varepsilon + \eta_x \geq -\varepsilon$ and

$$|V_1[i] - \hat{V}_2[i]| = |d_\infty(x_{i,1}, \Delta) - d_\infty(y_{i,2}, \Delta)| = |\eta_x + \alpha_2| \leq \varepsilon \leq 2\varepsilon.$$

- (iii) The minimum is the distance of a point to the diagonal for one term and the distance between the points for the other, say

$$\|x_{i,1} - x_{i,2}\|_\infty \leq \min\{d_\infty(x_{i,1}, \Delta), d_\infty(x_{i,2}, \Delta)\}$$

$$d_\infty(y_{i,1}, \Delta) \leq \min\{\|y_{i,1} - y_{i,2}\|_\infty, d_\infty(y_{i,2}, \Delta)\}$$

  Then $|V_1[i] - \hat{V}_2[i]| = |\|x_{i,1} - x_{i,2}\|_\infty - d_\infty(y_{i,1}, \Delta)|$. As $d_\infty(y_{i,1}, \Delta) \geq d_\infty(x_{i,1}, \Delta) - \varepsilon$, we have

$$\|x_{i,1} - x_{i,2}\|_\infty - d_\infty(y_{i,1}, \Delta) \leq \varepsilon + (\|x_{i,1} - x_{i,2}\|_\infty - d_\infty(x_{i,1}, \Delta)) \leq \varepsilon \leq 2\varepsilon$$

  We also have

$$d_\infty(y_{i,1}, \Delta) \leq \|y_{i,1} - y_{i,2}\|_\infty \leq \|x_{i,1} - x_{i,2}\|_\infty + 2\varepsilon,$$

  and thus

$$|V_1[i] - \hat{V}_2[i]| \leq 2\varepsilon.$$

Finally, $\|V_1 - \hat{V}_2\|_\infty \leq 2\varepsilon$. Now we prove and use the following lemma to conclude:

Lemma 6.3.3. *Let $D \in \mathbb{N}$ and $U, \hat{V} \in \mathbb{R}^D$. Assume that U is* non-increasing*: $\forall i, j \in \{1 \ldots n-1\}$, $i \leq j \Rightarrow U[i] \geq U[j]$. Let $V \in \mathbb{R}^D$ be the image of $\hat{V}$ by a coordinate permutation σ which makes it non-increasing: $\forall i, j \in \{1, \cdots, n-1\}$, $V[\sigma(i)] = \hat{V}[i]$ and $i \leq j \Rightarrow V[i] \geq V[i+1]$. Then:*

$$\|U - V\|_\infty \leq \|U - \hat{V}\|_\infty.$$

Proof. Let $\alpha = \|U - \hat{V}\|_\infty$. Let $i \in \{1, \cdots, n\}$ and $\hat{v}_i = \hat{V}[i] = U[i] + x_i$, where $-\alpha \leq x_i \leq \alpha$. Let σ be the coordinate permutation between V and $\hat{V}$, i.e. $\hat{v}_i = V[\sigma(i)]$. Let $m(i), M(i) \in \mathbb{N}$ be defined as:

$$M(i) = \min \{t : U[t] + \alpha < \hat{v}_i\}$$

(or $M(i) = n + 1$ if the set is empty) and

$$m(i) = \max \{t : U[t] - \alpha > \hat{v}_i\}$$

(or $m(i) = 0$ if the set is empty). Note that $m(i) < i < M(i)$ by definition. Since $t \leq m(i) \Rightarrow \hat{V}[t] > \hat{V}[i]$ and $t \geq M(i) \Rightarrow \hat{V}[t] < \hat{V}[i]$, there are at least $m(i)$ terms in $\hat{V}$ that are strictly larger than $\hat{v}_i$, and $D - M(i) + 1$ that are strictly smaller. Since V is non-increasing, it follows that:

$$m(i) + 1 \leq \sigma(i) \leq M(i) - 1.$$

Using the definition of $m(i)$, the fact that U is non-increasing and the equality $\hat{v}_i = U[i] + x_i$, we have $U[\sigma(i)] - U[i] \leq U[m(i)+1] - U[i] \leq \alpha + x_i$. Since $U[\sigma(i)] - V[\sigma(i)] = U[\sigma(i)] - \hat{V}[i] = U[\sigma(i)] - U[i] - x_i$, it follows that

$$U[\sigma(i)] - V[\sigma(i)] \leq \alpha.$$

Similarly, we have $U[\sigma(i)] - U[i] \geq U[M(i) - 1] - U[i] \geq x_i - \alpha$, and thus

$$U[\sigma(i)] - V[\sigma(i)] \geq -\alpha.$$

Finally, we have $|U[\sigma(i)] - V[\sigma(i)]| \leq \alpha$. This inequality being true for all i, it is also true for the vectors in the infinite norm and the proof is complete. □

We can finally conclude : $\|V_1 - V_2\|_\infty \leq \|V_1 - \hat{V}_2\|_\infty \leq 2\varepsilon$. □

Analysis of the stability bound. The dependence on N can lead to very small constants $C(N)$ in the worst case, which is not desireable as in practice. However, two remarks are worth considering at this point. Firstly, this constant disappears using the infinity norm, which is useful when using e.g. kNN classifiers. Secondly, this constant can be reduced by truncating the vectors, as stability is preserved whatever the number of components kept. In return, the vectors are less discriminative, so a trade-off has to be made in practice.

6.3.3 Application to 3D shape processing

In this section, we detail an application of persistence diagrams and corresponding topological vectors in 3D shape processing, in which descriptors are required to be Euclidean vectors. More precisely, we use persistence diagrams as *point descriptors* for *3D shape segmentation*.

Notation. We use *shape* as a shorthand for a compact smooth surface in $\mathbb{R}^3$.

Persistence diagrams as point descriptors. In order to provide a multiscale description of the structure of a shape X from the point of view of a single point $x \in X$, we consider the filtration induced by growing a geodesic ball centered at x, with radius r going from 0 to $+\infty$, i.e. the filtration induced by the sublevel sets of the distance function $f_x(\cdot) = d(x, \cdot)$—see Figures 6.9 and 6.10. We then encode the evolution of the ball's homology in the corresponding persistence diagram. Since we are dealing with surfaces, the 0-dimensional persistent homology is always trivial, whereas the 2-dimensional persistent homology has limited information (there is just one enclosed void, namely the surface itself). Hence, in practice, we compute the 1-dimensional persistence diagram and we add an extra point representing the unique 2-dimensional homological feature of the shape. This extra point has an infinite ordinate and an abscissa equal to the eccentricity of the source point. In particular, this allows distance-to-diagonal terms to naturally appear in the topological vector—see Definition 6.3.1. Finally, we also add the distance to the diagonal of this extra point in the topological vector since this does not affect its stability.

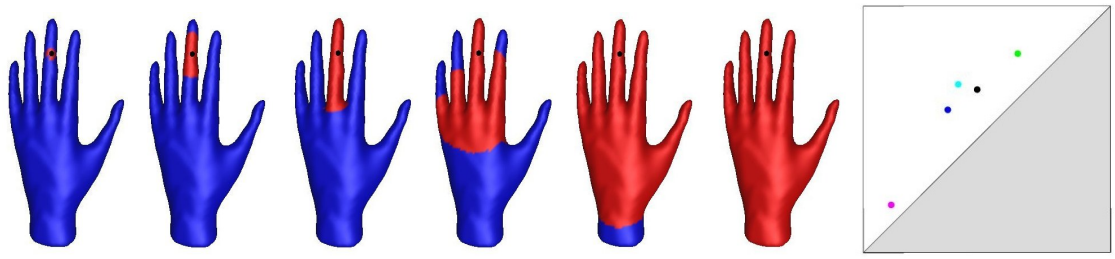

Figure 6.9: Geodesic balls centered at the black point are displayed in red. The persistence diagram corresponding to this family is shown in the far right. Note that each point can be easily associated with a shape part. The pink, blue, light blue, black and green points correspond to the middle, index, ring, pinky and thumb respectively. As the center point is close to the tip of the middle finger, one can see that its point in the persistence diagram is much closer to the diagonal than the other fingers. Note that for this shape, there are no essential holes.

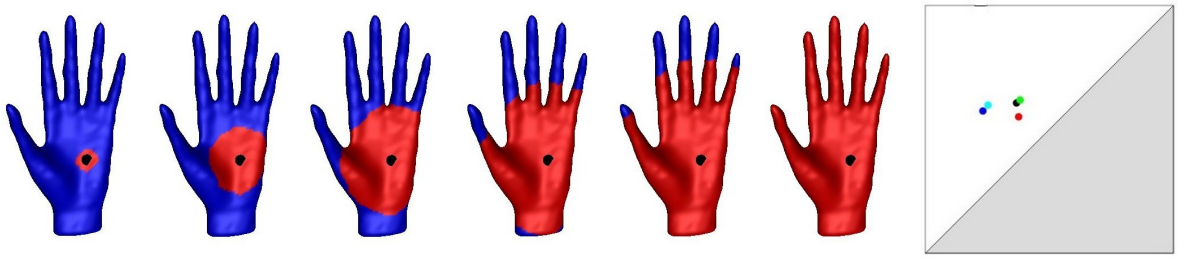

Figure 6.10: Same process as Figure 6.9 but with a different center point. Note the difference in the persistence diagram (far right). The colors in the diagram correspond to the same parts of the hand as in Figure 6.9. There is, however a new point in red, which corresponds to the hand base (palm), which was not present in the persistence diagram of the previous shape.

Distance to the diagonal. We also recall that the distance to the diagonal has a specific meaning. Indeed, if a point is very close to or is on the diagonal, it means that the corresponding hole was filled in quickly after being born in the growing process. In the case of 3D shape processing, this can be interpreted as a bump of small topographic prominence for instance, which can be considered as topological noise. The vertical distance of a point to the diagonal is exactly the prominence of the corresponding bump. On the contrary, the more salient a bump, the longer its prominence and thus the further away from the diagonal its point.

Example. We illustrate two such trackings for two different black center points in Figures 6.9 and 6.10. The growing process is shown from left to right with geodesic balls colored in red. If we consider Figure 6.9, we can see that in the first (left-most) image, the geodesic ball has no non-contractible cycles (holes) as it is simply connected. In the second image, the geodesic ball contains one inessential hole (at the tip of the middle finger). In the third one, there are no non-contractible holes again as the previous one is now filled in. In the fourth image, there are three inessential holes (the three other fingers). In the fifth one, there are no holes (notice that the thumb created a hole that was born and filled in between the fourth and fifth images). In the last image, the geodesic ball contains the entire shape, which has spherical topology and, as such, contains no essential holes. Therefore, the persistence diagram contains no points at infinity. Note that since the black base point is close to the tip of the middle finger, one of the points in the persistence diagram is both born and dead significantly earlier than the other ones.

Truncation level. The truncation level (or equivalently the dimension of the topological vectors) is driven by the prominent holes of each category (for instance this number would be 5 for a human

shape—two legs, two arms and the head—thus we would only keep around 5(5-1)/2=10 components in the vectors). In order to make the vectors independent of the scale, we consider the diagrams in log-scale (meaning that we apply the function $\log(1 + \cdot)$ on every birth and death value).

MDS and kNN. As an illustration, Figure 6.11 shows the topological vectors of all the points of a specific shape, plotted as points in $\mathbb{R}^3$ after a MultiDimensional Scaling (or MDS) on their distance matrix. The color of each vector is given by a ground truth segmentation provided with the input data set. Two remarks are in order at this stage: first, note that there is some continuity between vectors with identical labels, which suggests that the topological vectors vary continuously along the shape; second, and consequently, there is no natural grouping of the vectors into clusters, so unsupervised segmentation using traditional clustering algorithms such as k-means is likely to be ineffective. These observations suggest rather to use supervised learning algorithms in segmentation applications. We also show how such a kNN segmentation allows to achieve reasonable performance in Figure 6.12, even though the use of more elaborate algorithms like SVM leads to better results.

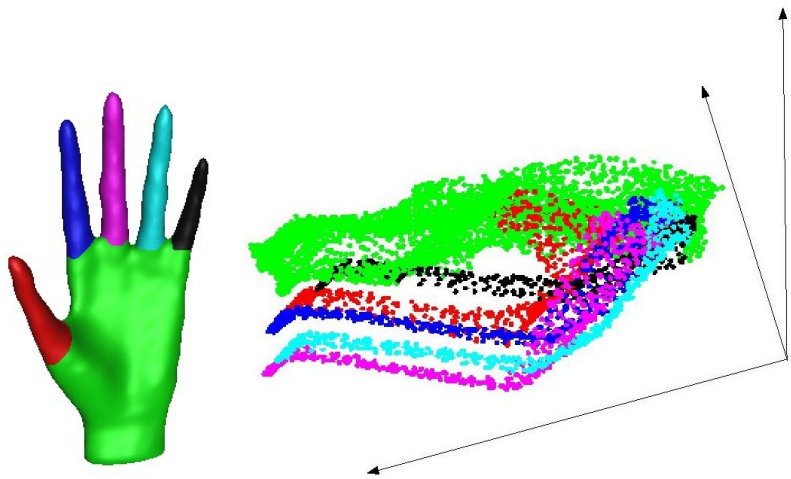

Figure 6.11: Example of MDS. One can easily observe the continuity between vectors of different labels. The color of each point refers to the same label as the colors displayed on the hand shape.

Stability. The main advantage of considering the persistence diagrams is that they enjoy stability properties, meaning that the difference between two diagrams cannot be too large if the they are computed from nearby points or on nearby shapes. This stability is a key feature in applications. It is stated formally in [37].

As an illustration of this stability property, we display components of topological vectors on shapes in various poses in Figure 6.13. Theorem 6.3.2 ensure that corresponding points have similar vectors. Note that the components of the topological vectors characterize parts of the shape that are difficult to relate to the other classical descriptors in the literature—apart from the first component, which roughly corresponds to the eccentricity—see the second paragraph of this section.

Computation. Unfortunately, 1-dimensional persistence is costly to compute in general. Indeed, if the shape is given by a triangle mesh with $O(m)$ edges and faces, the worst-case running time is of the order of $O(m^3)$ [103]. Note that this running time is the same for every center point. To overcome this difficulty in the case of 2D surfaces, we use Theorem 2.3.2, which states the equivalence between the inessential holes of the family of balls, i.e. points in $\mathrm{Ord}_1(f_x)$ and the inessential connected components (0-dimensional persistence) of the family of complements of these balls, i.e. points in $\mathrm{Ord}_0(-f_x)$. This means that, within every geodesic ball, every hole is associated to a connected component of the ball's complement. As connected components are much easier to track than holes (the complexity of computing 0-dimensional persistence diagrams is nearly linear), it is preferable to use them instead. Notice that, as we study the family of complements, the birth values are now bigger than the death ones (as the

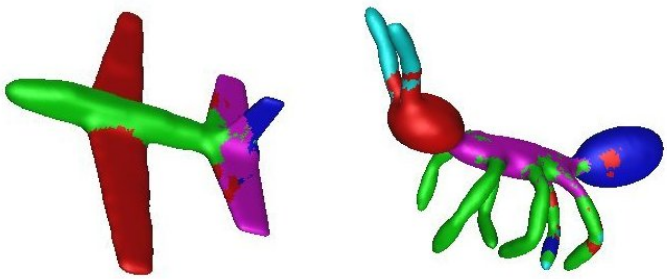

Figure 6.12: We compute the most common label for each face in a set of 100 nearest neighbors computed from a training set. No smoothing is applied but the segmentations on this pair of shapes are still reasonable (around 80 percent accuracy). However, this accuracy can decrease to 60 percent in other categories, thus we need a more elaborate algorithm for segmentation.

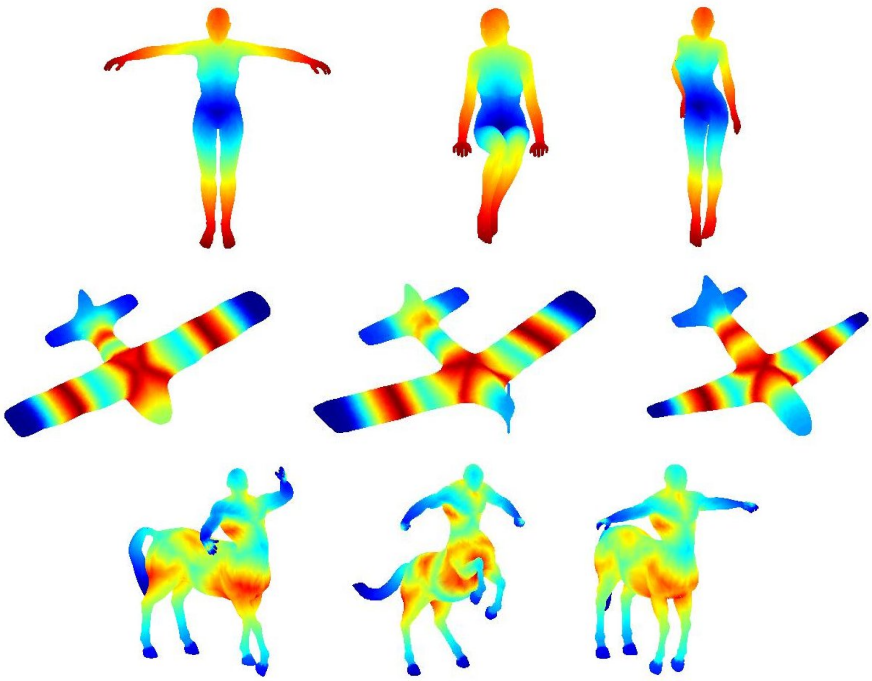

Figure 6.13: Topological vectors are computed on nearly isometric shapes. The first component is shown on the human shape, the second component is shown on the planes and the third one is shown on the centaurs. One can see that it respects the correspondence due to its stability.

radius is decreasing), leading to points under the diagonal. As an illustration, consider the family of the complements in Figure 6.9 (displayed in blue). Connected components of the blue sets are related to the holes of the red ones. However, note that Theorem 2.3.2 for essential holes, i.e. points in $\mathrm{Ext}_1(f_x)$, only associates them with essential holes of the complements, i.e. points in $\mathrm{Ext}_1(-f_x)$. The essential connected components of the family of complements of balls, i.e. points in $\mathrm{Ext}_0(f_x)$, are associated with the essential enclosed voids (2-dimensional topology) of the family of balls, i.e. points in $\mathrm{Ext}_2(-f_x)$, (see Figure 6.14). Thus, we cannot get access to the essential holes (the global loops or handles on the shape) with 0-dimensional persistence. This means that, although we gain a significant speedup in computational complexity, we lose some information when using duality, and in particular we do not track essential holes of 1-dimensional persistence.

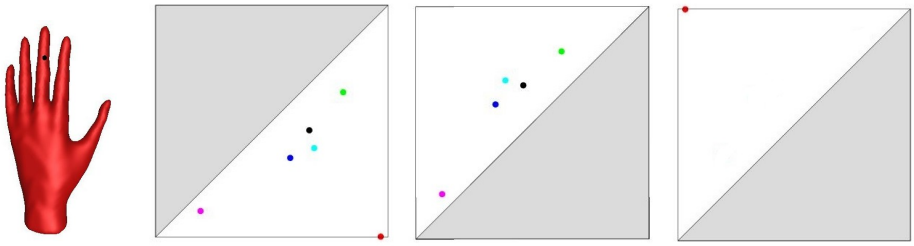

Figure 6.14: Left: base point shown in black. Middle: the 0, 1 and 2-dimensional persistence diagrams of the family of complements (0-dimensional) and the family of geodesic balls (1-dimensional and 2-dimensional). The symmetry theorem establishes the correspondence between the inessential points of the 0-dimensional and 1-dimensional persistence diagrams. They also match the essential point of the left-most persistence diagram (in red) with the essential point of the right-most persistence diagram. On this example, the 1-dimensional persistence diagram has no essential point, but if it had one, we would not be able to capture it in the 0-dimensional persistence diagram.

3D shape segmentation. In this paragraph, we use the topological vectors for supervised 3D shape segmentation and labeling. We use the Princeton benchmark [52] for both training and test shapes. This benchmark contains several different ground truth segmentations for each shape. On each shape that we use in the training set, we use the same ground truth segmentation as Kalogerakis et al. [87]. To show the improvement obtained when using our vector, we first consider the segmentation produced by using the method with 5 training shapes per category and the subset of features used in [87]. Table 6.4 (second column) shows an error percentage (computed with the Rand Index, which measures the segmentation quality as defined in [52], lower is better) obtained without using the topological vectors. In the same table (third column, S5+PDs) we report the error obtained by using the same pipeline, but augmented with the topological vectors, which on average has 15-20 dimensions. We recall—see the paragraph on symmetry—that the topological vectors cannot get access to essential hole (handles). This explains why the improvement is low in categories for which the segmentation characterizes handles (e.g. Cups). Other algorithms can be used to compute the full 1-dimensional homology [103] but they are more costly. We also believe that the bad result in the Glasses category is due to the fact that there are no prominent bumps on the Glasses shapes leading to nearly equal topological vectors almost everywhere that fool the classifier in the training process. Apart from that, note that in 18 out of 19 categories, we obtain an often significant improvement in the results. We also compare these results with the method of [87], which uses 6 and 19 training shapes (S6 and S19, respectively fourth and fifth columns of Table 6.4). Note that in 12 out of 19 categories our results are better than S6 and in 4 out of 19 categories better than S19, even though we used fewer training shapes, fewer features in each training shape, and no expensive penalty matrix optimization. Overall, this table shows that we can get close to the optimal results (where all-but-one shapes are used for training, leading to a huge amount of running time) with less data and features and demonstrate that topological vectors provides complementary information to the existing descriptors, and can potentially be useful in shape segmentation and labeling scenarios.

3D shape correspondence. We also use the topological vectors for shape matching. Since these vectors can be seen as a multivariate field defined on shapes, we decide to use the framework of functional map [114], and in particular the supervised learning approach. The exact procedure is fully described in [57]. We use 4 training shapes for several categories of the shape matching benchmark TOSCA [19] and compute optimal descriptor weights following the procedure described in [57]. We then use these weights to compute the optimal functional map on test shape pairs, by using 300 eigenvalues of the Laplace-Beltrami operator. We run this procedure two times to end up with two functional maps: one computed with the original set of classical probe functions (which includes all of the classical descriptors described in [87] plus more recent ones like HKS and WKS) and the other computed with the same set plus the topological vectors. We obtain large positive weights for the vectors, which indicates that it strongly influences the induced optimal functional map. Once the map is computed, it is also interesting to look at the induced correspondence. Figure 6.15 displays three error curves for every category. These

	S5	S5+PDs	S6	S19
Human	21.3	***11.3***	14.3	11.9
Cup	10.6	*10.1*	10.0	**9.9**
Glasses	21.8	*25.0*	14.1	**13.7**
Airplane	18.7	*9.3*	8.0	**7.9**
Ant	9.7	***1.5***	2.3	1.9
Chair	15.1	*7.3*	6.1	5.4
Octopus	5.5	*3.4*	2.2	1.8
Table	7.4	***2.5***	6.4	6.2
Teddy	6.0	*3.5*	5.3	**3.1**
Hand	21.1	*12.0*	13.9	**10.4**
Plier	12.3	*9.2*	10.0	**5.4**
Fish	20.9	***7.7***	14.2	12.9
Bird	24.8	*13.5*	14.8	**10.4**
Armadillo	18.4	*8.3*	8.4	**8.0**
Bust	35.4	22.0	33.4	**21.4**
Mech	22.7	*17.0*	12.7	**10.0**
Bearing	25.0	*11.2*	21.7	**9.7**
Vase	26.4	*17.8*	19.9	**16.0**
FourLeg	25.6	*15.8*	14.7	**13.7**

Table 6.4: Rand Indices computed over the segmentation benchmark. Results obtained with 5 training shapes without topological vectors (S5), and with them (S5+PDs), compared to results of Kalogerakis et al. [87] using significantly larger training sets (see text for details).

plots represent, given an unnormalized radius r, the percentage y of the points that are mapped by the correspondence at a distance at most r from their ground-truth image. One can see how the topological vectors strongly improve these error rates in all categories. We also show in Figure 6.16 the shape parts on which points get closer to their ground-truth image after adding the vectors. One can see that they correspond to flat, 'feature-less' parts of the shape, that are very difficult to characterize with classical descriptors whereas the multiscale definition of the topological vectors allows the corresponding probe functions to be much more discriminative.

6.4 Conclusion

In this chapter, we introduced the *Sliced Wasserstein kernel* and the *topological vectors*, which are two possible kernels for persistence diagrams that are provably *stable* with respect to $d_{\mathrm{w},1}$, the Sliced Wasserstein kernel being even equivalent to it for persistence diagrams with bounded cardinalities. We provided algorithms for computation, and we showed on several datasets substantial improvements in accuracy and training times (when tuning parameters is done with grid search) over competing kernels.

Metric properties of embeddings. Even though the Sliced Wasserstein kernel is provably equivalent to $d_{\mathrm{w},1}$, the lower bound depends on the number of diagram points, and converges to zero when the number of points increases. Our intuition is that this is the case for any mapping of persistence diagrams, i.e. either the upper or the lower bound depends on the number of points, and either converges to $+\infty$ (for the upper bound) or 0 (for the lower bound) when the number of points increases. Hence, we believe that a study about quantifying the metric distortion of a general mapping of persistence diagrams into a (possibly infinite dimensional) Hilbert space is possible and worth considering.

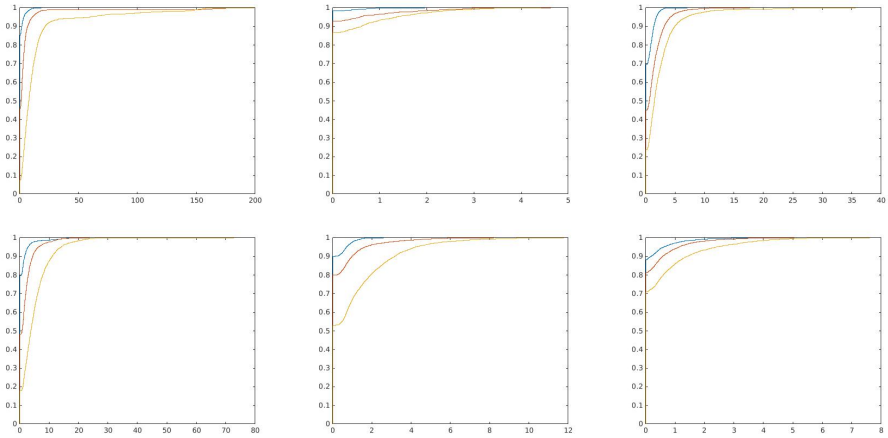

Figure 6.15: The blue curve represents the correspondence induced by the ground-truth functional map. The yellow one represents the correspondence induced by the optimal functional map without the topological vectors and the red one represents the correspondence induced by the optimal functional map with the vectors. The categories are, from left to right and top to bottom: horse, wolf, dog, cat, human and centaur.

Figure 6.16: Yellow parts are the ones which are the most improved with the topological vectors. Dark blue means no improvement. For every shape, it is clear: firstly that there is a positive improvement almost everywhere and secondly that the best improvements are obtained on the flat parts of the shapes.

CHAPTER 7

CONCLUSION

In this thesis, we presented several metric and statistical properties of topological descriptors. We showed that Reeb graphs can be compared efficiently with a pseudometric that is locally a true distance (Chapter 3), then, using this pseudometric, we proposed a way to metrize the space of Mappers in a stable way (Chapter 4), and we showed that the Mapper is an optimal estimator, for which we can compute confidence regions and automatic parameters (Chapter 5). Finally, we presented two methods to use the Mapper signatures, the persistence diagrams, in supervised Machine Learning (Chapter 6). In each chapter, we ended with some open questions raised by the chapter results.

More generally, the next long-term step for this work is to fit to current trends, both in Topological Data Analysis and in Machine Learning. Concerning topology, a lot of efforts is now devoted in the community to the extension of persistence theory to multivariate modules, i.e. vector spaces indexed by Euclidean vectors instead of the real line [23, 29, 53, 80]. In some cases, decomposition results exist, but stability is still unclear in general. Similarly, Reeb graphs and Mappers can be defined for multivariate functions, but our analysis in Chapters 3, 4 and 5 is anchored to functions that are real-valued.

Concerning supervised Machine Learning, the current hot topic is deep learning [77], whose gradient descent based algorithms optimize predictor functions depending on the architecture of a network of parameters. This field of Machine Learning achieved astounding results, i.e. in image classification and speech recognition, even though its theory is not well understood. Some works already did a first step towards the integration of topological descriptors into this field [25, 85, 95], but a comprehensive study on how this integration should be done is still lacking.

APPENDIX A

PROOF OF LEMMA 3.4.5

In this proof, we use the notations of Definition 2.3.3. Let

$$0 < \epsilon < \frac{1}{2} \min_{k=1,\dots,n} \min\{s_k - a_k, a_k - s_{k-1}\}.$$

The idea of the proof is to replace the right inverse of the projection $\pi : X \to \mathrm{R}_f(X)$ by a continuous map $\sigma : \mathrm{R}_f(X) \to X$ such that the composition $\pi \circ \sigma$ is homotopic to the identity of $\mathrm{R}_f(X)$. In order to make our new σ compatible with the function f, we need to perturb f to some other function g whose preimages of intervals $[s_i, s_j]$, $i \leq j$, are equal to the ones of f.

Let $g : X \to \mathbb{R}$ be defined by:

$$\forall x \in X,\ g(x) = \begin{cases} f(x) \text{ if } \min\limits_{k=1,\dots,n} |f(x) - a_k| > 2\epsilon \\ a_i \text{ otherwise, where } i = \mathrm{argmin}_k |f(x) - a_k| \end{cases}$$

As g is constant on equivalence classes of $\sim_f$, there is an induced map $\tilde{g} : \mathrm{R}_f(X) \to \mathbb{R}$. Moreover, for any $i \leq j$, we have $g^{-1}([s_i, s_j]) = f^{-1}([s_i, s_j])$ by definition of g and ϵ. The same holds for $\tilde{f}$ and $\tilde{g}$.

Now we want to define a continuous map $\sigma : \mathrm{R}_f(X) \to X$ such that the composition with the projection $\pi \circ \sigma$ is homotopic to $\mathrm{id}_{\mathrm{R}_f(X)}$. For any node v_i, if Y_{i-1} has k_i connected components $Y_{i-1}^1, \dots, Y_{i-1}^{k_i}$ and Y_i has l_i connected components $Y_i^1, \dots, Y_i^{l_i}$, we let $\{(\tilde{p}_{i-1}^k, p_{i-1}^k) \mid k = 1, \dots, k_i\}$ and $\{(q_i^l, \tilde{q}_i^l) \mid l = 1, \dots, l_i\}$ denote points in $\mathrm{R}_f(X)$ located at levelsets $a_i - 2\epsilon, a_i - \epsilon, a_i + \epsilon, a_i + 2\epsilon$. See Figure A.1. For any $i = 1, \dots, n$ and any $l = 1, \dots, l_i$, we select an arbitrary point $y_i^l \in Y_i^l$ and we let $s_i^l = \phi_i(y_i^l, a_i)$ and $\bar{s}_{i+1}^l = \psi_i(y_i^l, a_{i+1})$.

For any critical value a_i and any vertex v_i of $\mathrm{R}_f(X)$ at that level, we let $\sigma(v_i)$ be an arbitrary point in $\pi^{-1}(v_i)$, $\sigma(q_i^l) = s_i^l$, and $\sigma(p_{i-1}^k) = \bar{s}_i^k$. Moreover, as there exists a path $\gamma_k^{i,-} : [a_i - \epsilon, a_i] \to X$ from $\bar{s}_i^k$ to $\sigma(v_i)$, σ sends the arc $[p_{i-1}^k, v_i]$ to this path $\gamma_k^{i,-}$. Similarly, it sends the arc $[v_i, q_i^l]$ to a path $\gamma_l^{i,+} : [a_i, a_i + \epsilon] \to X$ from $\sigma(v_i)$ to s_i^l. Finally, σ also monotonically reparametrizes the arcs $[\tilde{p}_i^k, p_i^k]$ and $[q_i^l, \tilde{q}_i^l]$. Let $\mathrm{param}_i^+ : [a_i + \epsilon, a_i + 2\epsilon] \to [a_i, a_i + 2\epsilon]$, and $\mathrm{param}_i^- : [a_i - 2\epsilon, a_i - \epsilon] \to [a_i - 2\epsilon, a_i]$ be these reparametrizations. Again, see Figure A.1. More formally, let $x \in X$ and assume that $a_i \leq f(x) \leq a_{i+1}$ and that $\pi(x)$ belongs to the l-th edge of the Reeb graph between these two critical values. Then:

- $\sigma \circ \pi(x) = \mu_i(y_i^l, f(x))$ if $a_i + 2\epsilon \leq f(x) \leq a_{i+1} - 2\epsilon$;
- $\sigma \circ \pi(x) = \mu_i(y_i^l, \mathrm{param}_i^+ \circ f(x))$ if $a_i + \epsilon \leq f(x) \leq a_i + 2\epsilon$;
- $\sigma \circ \pi(x) = \mu_i(y_i^l, \mathrm{param}_{i+1}^- \circ f(x))$ if $a_{i+1} - 2\epsilon \leq f(x) \leq a_{i+1} - \epsilon$;
- $\sigma \circ \pi(x) = \gamma_l^{i,+}(f(x))$ if $a_i \leq f(x) \leq a_i + \epsilon$;
- $\sigma \circ \pi(x) = \gamma_l^{i+1,-}(f(x))$ if $a_{i+1} - \epsilon \leq f(x) \leq a_{i+1}$.

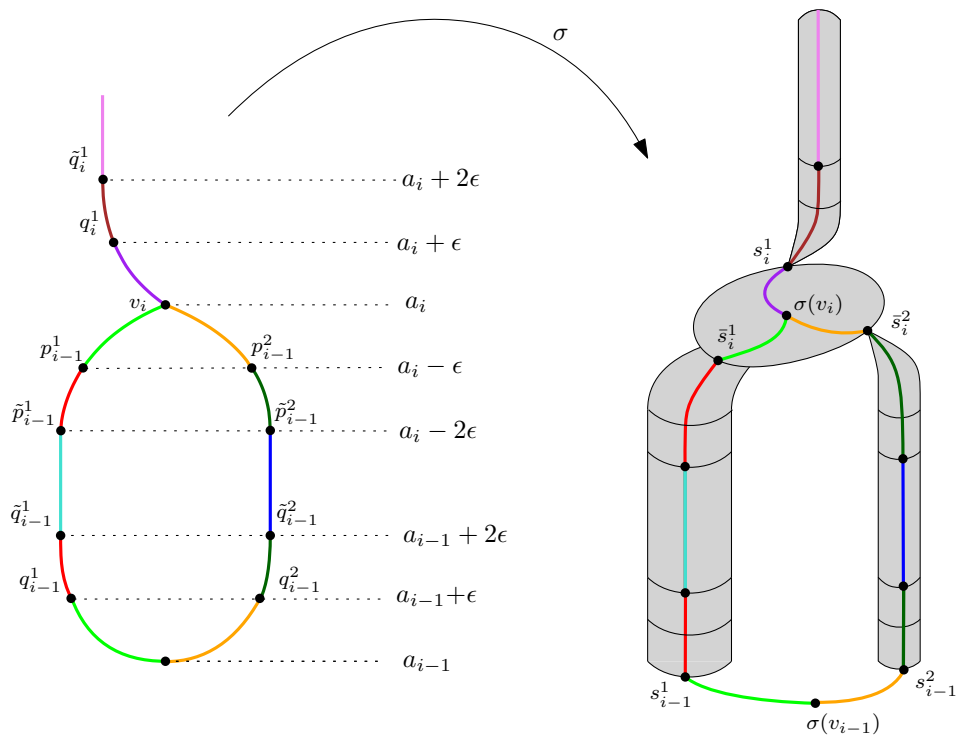

Figure A.1: The left panel displays the Reeb graph and the right panel displays the space X itself. σ sends an arc of the Reeb graph to the path with the same color in X.

By construction we have $g \circ \sigma = \tilde{g}$ and $\tilde{g} \circ \pi = g$ (note that this is not true for f).

Let $i \leq j$ and $I = [s_i, s_j]$. Then we have $\pi(g^{-1}(I)) \subseteq \tilde{g}^{-1}(I)$. Hence, π induces a morphism between $H_0(g^{-1}(I))$ and $H_0(\tilde{g}^{-1}(I))$. Let us show that this morphism is an isomorphism. Since π is surjective, this boils down to showing that x, y are connected in $g^{-1}(I)$ if and only if $\pi(x), \pi(y)$ are connected in $\tilde{g}^{-1}(I)$.

- If x, y are connected in $g^{-1}(I)$, then so are $\pi(x), \pi(y)$ in $\tilde{g}^{-1}(I)$, by continuity of π and the fact that $\tilde{g} \circ \pi = g$.
- If $\pi(x), \pi(y)$ are connected in $\tilde{g}^{-1}(I)$, then choose a path γ connecting $\pi(x)$ and $\pi(y)$. Now by definition of σ, there exists a path γ_x connecting x and $\sigma \circ \pi(x)$ in $g^{-1}(I)$. Indeed, σ can send $\pi(x)$ to five different locations in $g^{-1}(I)$ according to the value of $f(x)$, as seen above. Assume $f(x) \notin \mathrm{Crit}(f)$. Since there is a path $\tilde{\gamma}$ between x and $\mu_i(y_i^l, f(x))$, one can always find a path γ_x between x and $\sigma \circ \pi(x)$ in $g^{-1}(I)$ with an appropriate combination of $\tilde{\gamma}$, $\mu_i(y_i^l, \cdot)$ and $\gamma_l^{(i,+)/(i+1,-)}$. Now, assume $f(x) \in \mathrm{Crit}(f)$, and let $v_i = \pi(x)$. Then $\sigma(v_i)$ and x both belong to $\pi^{-1}(v_i)$, so they belong to the same connected component of the $g^{-1}(g(x))$ and one can find a path between them in $g^{-1}(I)$. Similarly, there exists a path γ_y connecting $\sigma \circ \pi(y)$ and y in $g^{-1}(I)$. Then $\gamma_y \circ \sigma(\gamma) \circ \gamma_x$ is a path between x and y in $g^{-1}(I)$ by continuity of σ and the fact that $g \circ \sigma = \tilde{g}$. So x, y are connected in $g^{-1}(I)$.

Since $g^{-1}(I) = f^{-1}(I)$ and $\tilde{g}^{-1}(I) = \tilde{f}^{-1}(I)$, we have that π_* is an isomorphism between $H_0(f^{-1}(I))$ and $H_0(\tilde{f}^{-1}(I))$, and the proof is complete.

BIBLIOGRAPHY

[1] Henry Adams, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. Persistence Images: A Stable Vector Representation of Persistent Homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.

[2] Pankaj Agarwal, Kyle Fox, Abhinandan Nath, Anastasios Sidiropoulos, and Yusu Wang. Computing the Gromov-Hausdorff Distance for Metric Trees. In *Proceedings of the 26th International Symposium on Algorithms and Computation*, 2015.

[3] Muthu Alagappan. From 5 to 13: Redefining the Positions in Basketball. MIT Sloan Sports Analytics Conference, 2012.

[4] Nachman Aronszajn. Theory of Reproducing Kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.

[5] Vincent Barra and Silvia Biasotti. 3D Shape Retrieval and Classification using Multiple Kernel Learning on Extended Reeb graphs. *The Visual Computer*, 30(11):1247–1259, 2014.

[6] Heinz Bauer. *Measure and integration theory*, volume 26 of *de Gruyter Studies in Mathematics*. Walter de Gruyter & Co., 2001.

[7] Ulrich Bauer, Barbara Di Fabio, and Claudia Landi. An Edit Distance for Reeb Graphs. In *Proceedings of the Eurographics 2016 Workshop on 3D Object Retrieval*, pages 27–34, 2016.

[8] Ulrich Bauer, Xiaoyin Ge, and Yusu Wang. Measuring Distance Between Reeb Graphs. In *Proceedings of the 30th Symposium on Computational Geometry*, pages 464–473, 2014.

[9] Ulrich Bauer, Xiaoyin Ge, and Yusu Wang. Measuring Distance between Reeb Graphs (v2). *CoRR*, abs/1307.2839v2, 2016.

[10] Ulrich Bauer, Elizabeth Munch, and Yusu Wang. Strong Equivalence of the Interleaving and Functional Distortion Metrics for Reeb Graphs. In *Proceedings of the 31st Symposium on Computational Geometry*, 2015.

[11] Christian Berg, Jens Christensen, and Paul Ressel. *Harmonic Analysis on Semigroups: Theory of Positive Definite and Related Functions*. Springer, 1984.

[12] Silvia Biasotti, Daniela Giorgi, Michela Spagnuolo, and Bianca Falcidieno. Reeb Graphs for Shape Analysis and Applications. *Theoretical Computer Science*, 392:5–22, 2008.

[13] Gérard Biau and André Mas. PCA-Kernel estimation. *Statistics and Risk Modeling with Applications in Finance and Insurance*, 29(1):19–46, 2012.

[14] Eckart Bindewald and Bruce Shapiro. Rna secondary structure prediction from sequence alignments using a network of k-nearest neighbor classifiers. *RNA*, 12(3):342–352, 2006.

[15] Håvard Bjerkevik. Stability of higher-dimensional interval decomposable persistence modules. *CoRR*, abs/1609.02086, 2016.

[16] Gilles Blanchard, Olivier Bousquet, and Laurent Zwald. Statistical properties of kernel principal component analysis. *Machine Learning*, 66(2-3):259–294, 2007.

[17] Andrew Blumberg, Itamar Gall, Michael Mandell, and Matthew Pancia. Robust Statistics, Hypothesis Testing, and Confidence Intervals for Persistent Homology on Metric Measure Spaces. *Foundations of Computational Mathematics*, 14:745–789, 2014.

[18] Magnus Botnan and Michael Lesnick. Algebraic Stability of Zigzag Persistence Modules. *CoRR*, abs/1604.00655, 2016.

[19] Alexander Bronstein, Michael Bronstein, and Ron Kimmel. *Numerical Geometry of Non-Rigid Shapes*. Springer, 2008.

[20] Peter Bubenik. Statistical Topological Data Analysis using Persistence Landscapes. *Journal of Machine Learning Research*, 16:77–102, 2015.

[21] Mickaël Buchet, Frédéric Chazal, Steve Oudot, and Donald Sheehy. Efficient and Robust Persistent Homology for Measures. In *Proceedings of the 26th Symposium on Discrete Algorithms*, pages 168–180, 2015.

[22] Dmitri Burago, Yuri Burago, and Sergei Ivanov. *A Course in Metric Geometry*. American Mathematical Society, 2001.

[23] Francesca Cagliari and Claudia Landi. Finiteness of rank invariants of multidimensional persistent homology groups. *Applied Mathematics Letters*, 24(4):516 – 518, 2011.

[24] Pablo Camara, Arnold Levine, and Raul Rabadan. Inference of Ancestral Recombination Graphs through Topological Data Analysis. *PLoS Computational Biology*, 12(8):1–25, 2016.

[25] Zixuan Cang and Guo-Wei Wei. TopologyNet: Topology Based Deep Convolutional and Multi-Task Neural Networks for Biomolecular Property Predictions. *PLoS Computational Biology*, 13(7), 2017.

[26] Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46:255–308, 2009.

[27] Gunnar Carlsson and Vin de Silva. Zigzag persistence. *Foundations of Computational Mathematics*, 10(4):367–405, 2010.

[28] Gunnar Carlsson, Vin de Silva, and Dmitriy Morozov. Zigzag Persistent Homology and Real-valued Functions. In *Proceedings of the 25th Symposium on Computational Geometry*, pages 247–256, 2009.

[29] Gunnar Carlsson and Afra Zomorodian. The Theory of Multidimensional Persistence. *Discrete and Computational Geometry*, 42(1):71–93, 2009.

[30] Mathieu Carrière. Cover complexes. In *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2017.

[31] Mathieu Carrière. Kernels for persistence diagrams. In *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2017.

[32] Mathieu Carrière, Marco Cuturi, and Steve Oudot. Sliced Wasserstein Kernel for Persistence Diagrams. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[33] Mathieu Carrière, Bertrand Michel, and Steve Oudot. Statistical Analysis and Parameter Selection for Mapper. *CoRR*, abs/1706.00204, 2017.

[34] Mathieu Carrière and Steve Oudot. Structure and Stability of the 1-Dimensional Mapper. *CoRR*, abs/1511.05823, 2015.

[35] Mathieu Carrière and Steve Oudot. Structure and Stability of the 1-Dimensional Mapper. In *Proceedings of the 32nd Symposium on Computational Geometry*, volume 51, pages 25:1–25:16, 2016.

[36] Mathieu Carrière and Steve Oudot. Local Equivalence and Induced Metrics for Reeb Graphs. In *Proceedings of the 33rd Symposium on Computational Geometry*, 2017.

[37] Mathieu Carrière, Steve Oudot, and Maks Ovsjanikov. Local Signatures using Persistence Diagrams. *HAL preprint*, 2015.

[38] Mathieu Carrière, Steve Oudot, and Maks Ovsjanikov. Stable Topological Signatures for Points on 3D Shapes. *Computer Graphics Forum*, 34, 2015.

[39] Joseph Chan, Gunnar Carlsson, and Raul Rabadan. Topology of viral evolution. *Proceedings of the National Academy of Science*, 110(46):18556–18571, 2013.

[40] C. Chang and C. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:1–27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[41] Amit Chattopadhyay, Hamish Carr, David Duke, Zhao Geng, and Osamu Saeki. Multivariate Topology Simplification. *Computational Geometry*, 58:1–24, 2016.

[42] Frédéric Chazal, David Cohen-Steiner, Marc Glisse, Leonidas Guibas, and Steve Oudot. Proximity of Persistence Modules and their Diagrams. In *Proceedings of the 25th Symposium on Computational Geometry*, pages 237–246, 2009.

[43] Frédéric Chazal, David Cohen-Steiner, Leonidas Guibas, Facundo Mémoli, and Steve Oudot. Gromov-Hausdorff Stable Signatures for Shapes using Persistence. *Computer Graphics Forum*, pages 1393–1403, 2009.

[44] Frédéric Chazal, David Cohen-Steiner, and Quentin Mérigot. Geometric Inference for Probability Measures. *Foundations of Computational Mathematics*, 11(6):733–751, 2011.

[45] Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Oudot. *The Structure and Stability of Persistence Modules*. Springer, 2016.

[46] Frédéric Chazal, Brittany Fasy, Fabrizio Lecci, Bertrand Michel, Alessandro Rinaldo, and Larry Wasserman. Robust topological inference: distance to a measure and kernel distance. *CoRR*, abs/1412.7197, 2014. Accepted for publication in Journal of Machine Learning Research.

[47] Frédéric Chazal, Brittany Fasy, Fabrizio Lecci, Bertrand Michel, Alessandro Rinaldo, and Larry Wasserman. Subsampling Methods for Persistent Homology. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2143–2151, 2015.

[48] Frédéric Chazal, Marc Glisse, Catherine Labruère, and Bertrand Michel. Optimal rates of convergence for persistence diagrams in Topological Data Analysis. *CoRR*, abs/1305.6239, 2013.

[49] Frédéric Chazal, Marc Glisse, Catherine Labruère, and Bertrand Michel. Convergence rates for persistence diagram estimation in topological data analysis. *Journal of Machine Learning Research*, 16:3603–3635, 2015.

[50] Frédéric Chazal, Leonidas Guibas, Steve Oudot, and Primoz Skraba. Analysis of scalar fields over point cloud data. In *Proceedings of the 20th Symposium on Discrete Algorithm*, pages 1021–1030, 2009.

[51] Frédéric Chazal, Pascal Massart, and Bertrand Michel. Rates of convergence for robust geometric inference. *Electronic Journal of Statistics*, 10(2):2243–2286, 2016.

[52] Xiaobai Chen, Aleksey Golovinskiy, and Thomas Funkhouser. A Benchmark for 3D Mesh Segmentation. *ACM Transactions on Graphics*, 28(3):1–12, 2009.

[53] Jérémy Cochoy and Steve Oudot. Decomposition of exact pfd persistence bimodules. *CoRR*, abs/1605.09726, 2016.

[54] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of Persistence Diagrams. *Discrete and Computational Geometry*, 37(1):103–120, 2007.

[55] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Extending persistence using Poincaré and Lefschetz duality. *Foundation of Computational Mathematics*, 9(1):79–103, 2009.

[56] Éric Colin de Verdière, Grégory Ginot, and Xavier Goaoc. Multinerves and Helly numbers of acyclic families. In *Proceedings of the 28th Symposium on Computational Geometry*, pages 209–218, 2012.

[57] Étienne Corman, Maks Ovsjanikov, and Antonin Chambolle. Supervised Descriptor Learning for Non-Rigid Shape Matching. In *Proceedings of the 6th Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment*, 2014.

[58] Antonio Cuevas. Set estimation: another bridge between statistics and geometry. *Boletín de Estadística e Investigación Operativa*, 25(2):71–85, 2009.

[59] Antonio Cuevas and Alberto Rodríguez-Casal. On boundary estimation. *Advances in Applied Probability*, pages 340–354, 2004.

[60] Navneet Dalal and Bill Triggs. Histograms of Oriented Gradients for Human Detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 886–893, 2005.

[61] Vin de Silva, Elizabeth Munch, and Amit Patel. Categorified Reeb Graphs. *Discrete and Computational Geometry*, 55:854–906, 2016.

[62] Ronald DeVore and George Lorentz. *Constructive approximation*, volume 303. Springer Science & Business Media, 1993.

[63] Tamal Dey, Fengtao Fan, and Yusu Wang. Graph Induced Complex on Point Data. In *Proceedings of the 29th Symposium on Computational Geometry*, pages 107–116, 2013.

[64] Tamal Dey and Yusu Wang. Reeb Graphs: Approximation and Persistence. *Discrete and Computational Geometry*, 49(1):46–73, 2013.

[65] Barbara di Fabio and Massimo Ferri. Comparing persistence diagrams through complex vectors. *CoRR*, abs/1505.01335, 2015.

[66] Barbara di Fabio and Claudia Landi. The Edit Distance for Reeb Graphs of Surfaces. *Discrete and Computational Geometry*, 55(2):423–461, 2016.

[67] Pawel Dlotko. Cubical complex. In *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2015.

[68] Herbert Edelsbrunner and John Harer. Jacobi sets of multiple Morse functions. *Foundations of Computational Mathematics*, pages 37–57, 2002.

[69] Herbert Edelsbrunner and John Harer. *Computational Topology: an introduction*. AMS Bookstore, 2010.

[70] Herbert Edelsbrunner, John Harer, and Amit Patel. Reeb Spaces of Piecewise Linear Mappings. In *Proceedings of the 24th Symposium on Computational Geometry*, pages 242–250, 2008.

[71] Brittany Fasy, Fabrizio Lecci, Alessandro Rinaldo, Larry Wasserman, Sivaraman Balakrishnan, and Aarti Singh. Confidence Sets for Persistence Diagrams. *The Annals of Statistics*, 42(6):2301–2339, 2014.

[72] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning*. Springer series in statistics Springer, 2001.

[73] Marcio Gameiro, Yasuaki Hiraoka, and Ippei Obayashi. Continuation of Point Clouds via Persistence Diagrams. *Physica D: Nonlinear Phenomena*, 334:118–132, 2016.

[74] Xiaoyin Ge, Issam Safa, Mikhail Belkin, and Yusu Wang. Data Skeletonization via Reeb Graphs. In *Advances in Neural Information Processing Systems 24*, pages 837–845, 2011.

[75] Christopher Genovese, Marco Perone-Pacifico, Isabella Verdinelli, and Larry Wasserman. Manifold estimation and singular deconvolution under Hausdorff loss. *The Annals of Statistics*, 40:941–963, 2012.

[76] Christopher Genovese, Marco Perone-Pacifico, Isabella Verdinelli, and Larry Wasserman. Minimax Manifold Estimation. *Journal of Machine Learning Research*, 13:1263–1291, 2012.

[77] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[78] Sara Goodwin, John McPherson, and Richard McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nature Review Genetics*, 17(6):333–351, 2016.

[79] Zhenhua Guo, Lei Zhang, and David Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE Transaction on Image Processing*, pages 1657–1663, 2010.

[80] Heather Harrington, Nina Otter, Hal Schenck, and Ulrike Tillmann. Stratifying multiparameter persistent homology. *CoRR*, abs/1708.07390, 2017.

[81] William Harvey, Yusu Wang, and Rephael Wenger. A randomized O(m log m) time algorithm for computing Reeb graphs of arbitrary simplicial complexes. In *Proceedings of the 26th Symposium on Computational Geometry*, pages 267–276, 2010.

[82] Jan-Martin Hertzsch, Rob Sturman, and Stephen Wiggins. DNA microarrays: design principles for maximizing ergodic, chaotic mixing. In *Small*, volume 3, pages 202–218, 2007.

[83] TS. Hinks, X. Zhou, KJ. Staples, BD. Dimitrov, A. Manta, T. Petrossian, P. Lum, CG. Smith, JA. Ward, PH Howarth, AF. Walls, SD. Gadola, and R. Djukanovic. Innate and adaptive t cells in asthmatic patients: Relationship to severity and disease mechanisms. *Journal of Allergy and Clinical Immunology*, 136(2):323–333, 2015.

[84] Yasuaki Hiraoka, Takenobu Nakamura, Akihiko Hirata, Emerson Escolar, Kaname Matsue, and Yasumasa Nishiura. Hierarchical structures of amorphous solids characterized by persistent homology. In *Proceedings of the National Academy of Science*, volume 26, 2016.

[85] Christoph Hofer, Roland Kwitt, Marc Niethammer, and Andreas Uhl. Deep Learning with Topological Signatures. *CoRR*, abs/1707.04041, 2017. Accepted to Advances in Neural Information Processing Systems 30.

[86] Andrew Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433–449, 1999.

[87] Evangelos Kalogerakis, Aaron Hertzmann, and Karan Singh. Learning 3D Mesh Segmentation and Labeling. *ACM Transactions on Graphics*, 29(4):102, 2010.

[88] Min-su Kim, Benjamin Hur, and Sun Kim. Rddpred: a condition-specific rna-editing prediction model from rna-seq data. *BMC Genomics*, 17(1), 2016.

[89] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[90] Genki Kusano, Kenji Fukumizu, and Yasuaki Hiraoka. Persistence Weighted Gaussian Kernel for Topological Data Analysis. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 2004–2013, 2016.

[91] Genki Kusano, Kenji Fukumizu, and Yasuaki Hiraoka. Kernel method for persistence diagrams via kernel embedding and weight factor. *CoRR*, abs/1706.03472, 2017.

[92] Roland Kwitt, Stefan Huber, Marc Niethammer, Weili Lin, and Ulrich Bauer. Statistical Topological Data Analysis - A Kernel Perspective. In *Advances in Neural Information Processing Systems 28*, pages 3070–3078, 2015.

[93] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[94] Chunyuan Li, Maks Ovsjanikov, and Frédéric Chazal. Persistence-Based Structural Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2003–2010, 2014.

[95] Jen-Yu Liu, Shyh-Kang Jeng, and Yi-Hsuan Yang. Applying Topological Persistence in Convolutional Neural Network for Music Audio Signals. *CoRR*, abs/1608.07373, 2016.

[96] David Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[97] P. Lum, G. Singh, A. Lehman, T. Ishkanov, M. Vejdemo-Johansson, M. Alagappan, J. Carlsson, and G. Carlsson. Extracting insights from the shape of complex data using topology. *Scientific Reports*, 3, 2013.

[98] Stéphane Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 2008.

[99] Clément Maria. Persistent cohomology. In *GUDHI User and Reference Manual.* GUDHI Editorial Board, 2015.

[100] Michael Metzker. Sequencing technologies - the next generation. *Nature Review Genetics*, 11(1):31–46, 2010.

[101] Waleed Mohamed and A. Ben Hamza. Reeb graph path dissimilarity for 3d object matching and retrieval. *The Visual Computer*, 28(3):305–318, 2012.

[102] Vlad Morariu, Balaji Srinivasan, Vikas Raykar, Ramani Duraiswami, and Larry Davis. Automatic online tuning for fast Gaussian summation. In *Advances in Neural Information Processing Systems 21*, pages 1113–1120, 2009.

[103] Dmitriy Morozov. *Homological Illusions of Persistence and Stability.* Ph.D. dissertation, Department of Computer Science, Duke University, 2008.

[104] Tomoyuki Mukasa, Shohei Nobuhara, Atsuto Maki, and Takashi Matsuyama. Finding articulated body in time-series volume data. In *Proceedings of the 4th International Conference on Articulated Motion and Deformable Objects*, pages 395–404, 2006.

[105] Elizabeth Munch and Bei Wang. Convergence between Categorical Representations of Reeb Space and Mapper. In *Proceedings of the 32nd Symposium on Computational Geometry*, volume 51, pages 53:1–53:16, 2016.

[106] James Munkres. *Elements of Algebraic Topology.* Addison-Wesley, 1993.

[107] Sameer Nene, Shree Nayar, and Hiroshi Murase. Columbia Object Image Library (COIL-100). Technical Report CUCS-006-96, 1996.

[108] Monica Nicolau, Arnold Levine, and Gunnar Carlsson. Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proceedings of the National Academy of Science*, 108(17):7265–7270, 2011.

[109] Jessica Nielson, Jesse Paquette, Aiwen Liu, Cristian Guandique, Amy Tovar, Tomoo Inoue, Karen-Amanda Irvine, John Gensel, Jennifer Kloke, Tanya Petrossian, Pek Lum, Gunnar Carlsson, Geoffrey Manley, Wise Young, Michael Beattie, Jacqueline Bresnahan, and Adam Ferguson. Topological data analysis for discovery in preclinical spinal cord injury and traumatic brain injury. *Nature Communications*, 6, 2015.

[110] Shin-Ichi Ohta. Gradient flows on Wasserstein spaces over compact Alexandrov spaces. *American Journal Mathematics*, 131(2):475–516, 2009.

[111] Shin-Ichi Ohta. Barycenters in Alexandrov spaces of curvature bounded below. *Advances in Geometry*, 12:571–587, 2012.

[112] Timo Ojala, Topi Mäenpää, Matti Pietikäinen, Jaakko Viertola, Juha Kyllönen, and Sami Huovinen. Outex - new framework for empirical evaluation of texture analysis algorithms. In *Proceedings of the 16th International Conference on Pattern Recognition*, pages 701–706, 2002.

[113] Steve Oudot. *Persistence Theory: From Quiver Representations to Data Analysis.* Number 209 in Mathematical Surveys and Monographs. American Mathematical Society, 2015.

[114] Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics*, 31(4):30, 2012.

[115] Valerio Pascucci, Giorgio Scorzelli, Peer-Timo Bremer, and Ajith Mascarenhas. Robust On-line Computation of Reeb Graphs: Simplicity and Speed. In *Proceedings of ACM SIGGRAPH 2007*, 2007.

[116] Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 435–446, 2011.

[117] Ali Rahimi and Benjamin Recht. Random Features for Large-Scale Kernel Machines. In *Advances in Neural Information Processing Systems 20*, pages 1177–1184, 2008.

[118] Gerald Reaven and Rupert Miller. An attempt to define the nature of chemical diabetes using a multidimensional analysis. *Diabetologia*, 16(1):17–24, 1979.

[119] Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. A Stable Multi-Scale Kernel for Topological Machine Learning. *CoRR*, abs/1412.6821, 2014.

[120] Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. A Stable Multi-Scale Kernel for Topological Machine Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[121] Vanessa Robins and Katharine Turner. Principal Component Analysis of Persistent Homology Rank Functions with case studies of Spatial Point Patterns, Sphere Packing and Colloids. *Physica D: Nonlinear Phenomena*, 334:1–186, 2016.

[122] Matteo Rucco, Emanuela Merelli, Damir Herman, Devi Ramanan, Tanya Petrossian, Lorenzo Falsetti, Cinzia Nitti, and Aldo Salvi. Using topological data analysis for diagnosis pulmonary embolism. *Journal of Theoretical and Applied Computer Science*, 9(1):41–55, 2015.

[123] Bryan Russell, Antonio Torralba, Kevin Murphy, and William Freeman. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008.

[124] Filippo Santambrogio. *Optimal transport for applied mathematicians*. Springer, 2015.

[125] G. Sarikonda, J. Pettus, S. Phatak, S. Sachithanantham, JF. Miller, JD. Wesley, E. Cadag, J. Chae, L. Ganesan, R. Mallios, S. Edelman, B. Peters, and M. von Herrath. Cd8 t-cell reactivity to islet antigens is unique to type 1 while cd4 t-cell reactivity exists in both type 1 and type 2 diabetes. *Journal of Autoimmunity*, 50:77–82, 2014.

[126] Bernhard Schölkopf, Ralf Herbrich, and Alex Smola. A Generalized Representer Theorem. In *Proceedings of the 14th Annual Conference on Computational Learning Theory*, pages 416–426, 2001.

[127] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[128] John Shawe-Taylor, Christopher Williams, Nello Cristianini, and Jaz Kandola. On the eigenspectrum of the Gram matrix and the generalization error of kernel-PCA. *IEEE Transactions on Information Theory*, 51(7):2510–2522, 2005.

[129] Gurjeet Singh, Facundo Mémoli, and Gunnar Carlsson. Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition. In *Symposium on Point Based Graphics*, pages 91–100, 2007.

[130] George Soumya and Joseph Shibily. Text Classification by Augmenting Bag of Words (BOW) Representation with Co-occurrence Feature. *IOSR Journal of Computer Engineering*, 16:34–38, 2014.

[131] Wilson Sutherland. *Introduction to Metric and Topological Spaces*. Oxford University Press, 2009.

[132] Joshua Tenenbaum, Vin de Silva, and John Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500), 2000.

[133] Julien Tierny and Hamish Carr. Jacobi Fiber Surfaces for Bivariate Reeb Space Computation. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):960–969, 2017.

[134] Julien Tierny, David Guenther, and Valerio Pascucci. Optimal General Simplification of Scalar Fields on Surfaces. In *Topological and Statistical Methods for Complex Data*. 2015.

[135] Julien Tierny, Jean-Philippe Vandeborre, and Mohamed Daoudi. Invariant High Level Reeb Graphs of 3D Polygonal Meshes. *International Symposium on 3D Data Processing Visualization and Transmission*, pages 105–112, 2006.

[136] Katharine Turner, Yuriy Mileyko, Sayan Mukherjee, and John Harer. Fréchet Means for Distributions of Persistence Diagrams. *Discrete and Computational Geometry*, 52(1):44–70, 2014.

[137] Cédric Villani. *Optimal transport : old and new.* Springer, 2009.

[138] Yuan Yao, Jian Sun, Xuhui Huang, Greg Bowman, Gurjeet Singh, Michael Lesnick, Leonidas Guibas, Vijay Pande, and Gunnar Carlsson. Topological methods for exploring low-density states in biomolecular folding pathways. *Journal of Chemical Physics*, 130(14), 2009.

[139] Bin Yu. Assouad, Fano, and Le Cam. In *Festschrift for Lucien Le Cam*, pages 423–435. Springer, 1997.

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : On metric and statistical properties of topological descriptors for geometric data

Mots clefs : Analyse de données topologiques, Statistique, Apprentissage automatique, Méthodes à noyaux

Résumé : Dans le cadre de l'apprentissage automatique, l'utilisation de représentations alternatives, ou descripteurs, pour les données est un problème fondamental permettant d'améliorer sensiblement les résultats des algorithmes. Parmi eux, les descripteurs topologiques calculent et encodent l'information de nature topologique contenue dans les données géométriques. Ils ont pour avantage de bénéficier de nombreuses bonnes propriétés issues de la topologie, et désirables en pratique, comme par exemple leur invariance aux déformations continues des données. En revanche, la structure et les opérations nécessaires à de nombreuses méthodes d'apprentissage, comme les moyennes ou les produits scalaires, sont souvent absents de l'espace de ces descripteurs. Dans cette thèse, nous étudions en détail les propriétés métriques et statistiques des descripteurs topologiques les plus fréquents, à savoir les diagrammes de persistance et Mapper. En particulier, nous montrons que le Mapper, qui est empiriquement un descripteur instable, peut être stabilisé avec une métrique appropriée, que l'on utilise ensuite pour calculer des régions de confiance et pour régler automatiquement ses paramètres. En ce qui concerne les diagrammes de persistance, nous montrons que des produits scalaires peuvent être utilisés via des méthodes à noyaux, en définissant deux noyaux, ou plongements, dans des espaces de Hilbert en dimension finie et infinie.

Title : On Metric and Statistical Properties of Topological Descriptors for Geometric Data

Keywords : Topological Data Analysis, Statistics, Unsupervised Machine Learning, Kernel Methods

Abstract : In the context of supervised Machine Learning, finding alternate representations, or descriptors, for data is of primary interest since it can greatly enhance the performance of algorithms. Among them, topological descriptors focus on and encode the topological information contained in geometric data. One advantage of using these descriptors is that they enjoy many good and desirable properties, due to their topological nature. For instance, they are invariant to continuous deformations of data. However, the main drawback of these descriptors is that they often lack the structure and operations required by most Machine Learning algorithms, such as a means or scalar products. In this thesis, we study the metric and statistical properties of the most common topological descriptors, the persistence diagrams and the Mappers. In particular, we show that the Mapper, which is empirically instable, can be stabilized with an appropriate metric, that we use later on to compute confidence regions and automatic tuning of its parameters. Concerning persistence diagrams, we show that scalar products can be defined with kernel methods by defining two kernels, or embeddings, into finite and infinite dimensional Hilbert spaces.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France