

Separating Features from Noise with Persistence and Statistics

by

Bei Wang

Department of Computer Science
Duke University

Date: _____

Approved:

Herbert Edelsbrunner, Advisor

Pankaj K. Agarwal

Kamesh Munagala

Sayan Mukherjee

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Computer Science
in the Graduate School of Duke University
2010

ABSTRACT
(Computer Science)

Separating Features from Noise
with Persistence and Statistics

by

Bei Wang

Department of Computer Science
Duke University

Date: _____

Approved:

Herbert Edelsbrunner, Advisor

Pankaj K. Agarwal

Kamesh Munagala

Sayan Mukherjee

An abstract of a dissertation submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy in the Department of Computer Science
in the Graduate School of Duke University
2010

Abstract

In this thesis, we explore techniques in statistics and persistent homology, which detect features among data sets such as graphs, triangulations and point cloud. We accompany our theorems with algorithms and experiments, to demonstrate their effectiveness in practice.

We start with the derivation of graph scan statistics, a measure useful to assess the statistical significance of a subgraph in terms of edge density. We cluster graphs into densely-connected subgraphs based on this measure. We give algorithms for finding such clusterings and experiment on real-world data.

We next study statistics on persistence, for piecewise-linear functions defined on the triangulations of topological spaces. We derive persistence pairing probabilities among vertices in the triangulation. We also provide upper bounds for total persistence in expectation.

We continue by examining the elevation function defined on the triangulation of a surface. Its local maxima obtained by persistence pairing are useful in describing features of the triangulations of protein surfaces. We describe an algorithm to compute these local maxima, with a run-time ten-thousand times faster in practice than previous method. We connect such improvement with the total Gaussian curvature of the surfaces.

Finally, we study a stratification learning problem: given a point cloud sampled from a stratified space, which points belong to the same strata, at a given scale level? We assess the local structure of a point in relation to its neighbors using kernel and cokernel

persistent homology. We prove the effectiveness of such assessment through several inference theorems, under the assumption of dense sample. The topological inference theorem relates the sample density with the homological feature size. The probabilistic inference theorem provides sample estimates to assess the local structure with confidence. We describe an algorithm that computes the kernel and cokernel persistence diagrams and prove its correctness. We further experiment on simple synthetic data.

To all the ones I love.

Contents

Abstract	iv
List of Tables	x
List of Figures	xi
Acknowledgements	xiv
1 Introduction	1
2 Spatial Scan Statistics for Graph Clustering	6
2.1 Introduction	6
2.2 Preliminaries	9
2.3 Graph Scan Statistic	20
2.3.1 Properties of Graph Scan Statistics	25
2.3.2 Graph Scan Statistics and Local Modularity	26
2.3.3 Graph Scan Statistics and Bregman Divergences	28
2.4 Algorithms	30
2.5 Analysis	34
2.6 Conclusions	40
Appendix to Chapter 2	42
A Proofs for the Properties of Graph Scan Statistic	42
B Approximation of d_P	44

3	Persistence in Expectation	47
3.1	Introduction	47
3.2	Preliminaries	49
3.3	Pairing Probabilities	54
3.4	Expected Total Persistence	58
3.5	Change in Total Persistence	66
3.6	Discussion	68
4	Elevation	70
4.1	Introduction	70
4.2	Preliminaries	72
4.3	Computation	81
4.4	Experiments	86
4.5	Conclusion and Discussion	90
5	Towards Stratification Learning through Homology Inference	91
5.1	Introduction	91
5.2	Background	93
5.2.1	Persistence Modules	93
5.2.2	Homology	97
5.2.3	Stratified Spaces	99
5.3	Topological Inference Theorem	102
5.3.1	Local Equivalence	103
5.3.2	Inference Theorem	106
5.4	Geometric Lower Bound	112
5.4.1	Absolute Homology Modules	112
5.4.2	Geometric Lower Bounds	113

5.5	Probabilistic Inference Theorem	116
5.6	Algorithm	120
5.6.1	Clustering	121
5.6.2	Diagram Computation	121
5.6.3	Correctness	127
5.7	Simulations	128
5.8	Discussion	129
	Appendix to Chapter 5	133
A	Defining the Map ϕ	133
A.1	Background	133
A.2	Intersection Map Details	135
B	Control of η Parameters	139
C	Algorithm Details	141
D	Algorithmic Correctness	144
D.1	Bottom Face	145
D.2	Top Face	147
D.3	Left and Right Faces	149
D.4	Finale	159
6	Discussion	160
	Bibliography	163
	Glossary	173
	Biography	185

List of Tables

2.1	Sizes of real-world datasets and the average runtime.	35
2.2	Power of Greedy Nibble and Greedy Chomp.	36
2.3	d_P values of top clusters found with MCL and Ncut.	37
2.4	Cluster overlap analysis.	37
3.1	Classification of the vertices in a PL function on a 2-manifold.	51
3.2	Examples of Q	58
4.1	Information on the triangulated surfaces used in the experiments.	87
4.2	Statistics on the number of critical points and critical regions.	88
4.3	Statistics on box intersections and critical region intersections.	88
4.4	Dominant terms in the running time analysis.	89

List of Figures

1.1	Examples of various data types: graphs, triangulations and point cloud. . .	2
2.1	Example of a cluster and its various measures.	15
2.2	Intuitive understanding of likelihood and probability.	17
2.3	Type I and Type II error.	19
2.4	Comparison of d_P with modularity.	27
2.5	Plot of $1/\gamma$ vs. average cluster size.	38
2.6	Cluster rank vs. cluster discrepancy.	39
2.7	Compare d_P with Q_1 and Q_2	45
2.8	Compare p -value of d_P with Q_1 and Q_2	46
3.1	Examples of the lower star and lower link.	51
3.2	Two cases in the proof of pairing probabilities.	55
3.3	Symmetric cases in the proof of pairing probabilities.	57
3.4	Sort vertices in the lower link.	61
3.5	Sort vertices in the lower link.	63
3.6	Relative multiplicities between intervals.	64
4.1	From left to right: a minimum, a saddle, and a maximum of the vertical height function.	73
4.2	Example of an elevation function defined in a given direction.	75
4.3	The four generic types of local maxima of the elevation function.	77
4.4	Critical regions of maximum, minimum and monkey saddle.	81

4.5	Data representative 1BRS protein peptide.	87
5.1	Persistence modules.	94
5.2	Commuting diagrams for strongly interleaving persistence modules.	95
5.3	Illustration of a point cloud and its persistence diagram.	96
5.4	Persistence diagram for relative homology.	99
5.5	Example of the stratification of a pinched torus with a spanning disc stretched across the hole.	99
5.6	A 2-dimensional stratified space and corn construction.	100
5.7	Illustration of equivalence relation.	104
5.8	Regions in \mathbb{X} -diagrams and U -diagrams.	108
5.9	Empty rectangle in the ker/cok persistence diagrams.	108
5.10	Kernel persistence diagram of two local equivalent points, given \mathbb{X}	110
5.11	Kernel persistence diagram of two local equivalent points, given U	110
5.12	Kernel persistence diagram of two points that are not local equivalent, given \mathbb{X}	111
5.13	Kernel persistence diagram of two points that are not local equivalent, given U	111
5.14	Illustration of $Z(\alpha)$	124
5.15	Illustration of the lune and the moon.	125
5.16	Illustration of the simplicial complexes constructed.	126
5.17	Examples of simple synthetic data.	128
5.18	Points sampled from a cross and their corresponding ker/cok persistence diagrams.	129
5.19	Points sampled from a plane intersecting a line, and their corresponding ker/cok persistence diagrams.	130
5.20	Points sampled from two intersecting planes, and their corresponding ker/cok persistence diagrams.	131
5.21	Definition of j , an example.	136

5.22	Cases in the proof of Proposition B.2.	141
5.23	An example of the implicit perturbation.	144
5.24	Two adjacent commuting cubes.	145
5.25	Map j' and f' for a linear singular simplex.	151
5.26	Map j' for a linear singular simplex that requires barycentric subdivision.	152
5.27	Case (D.1.a): illustration of F	154
5.28	Case (D.1.c): illustration of F	155
5.29	Illustration of G	156

Acknowledgements

I want to thank my advisor Herbert Edelsbrunner, for his guidance, patience and inspiration. When I started working on the Elevation function, during the many meetings in his office, I had quite a few *Déjà vu* moments when U2 kept singing in my head: "... A star, lit up like a cigar, strung out like a guitar, maybe you can educate my mind..." Herbert did just that.

I would like to thank my committee member Sayan Mukherjee, for his support and pointed advise. I would also like to thank the rest of my committee members, Pankaj Agarwal and Kamesh Munagala, for their comments and suggestions towards my dissertation.

I would like to thank Diane Riggs for her help and support during my entire graduate study. She made my graduate school life smooth and worry-free.

I would like to thank my collaborators, coauthors and colleagues, many of them are my friends as well. I specifically thank several of my coauthors and collaborators for great conversations and discussions. In no particular order, they are: Jeff Phillips, Paul Bendich, Robert Schrieber, Dennis Wilkinson, Nina Mishra, Robert Tarjan, Sudheer Sahu, John Reif, Dmitriy Morozov, Terrence Furey, Dimitris Papamichail, Steven Skiena, Mikael Vejdemo-Johansson and Randolf Rotta. I would like to thank those who have provided great feedback and inspiration during my research. To list a few: Yuriy Mileyko, Chao Chen, Amit Patel, Brittany Fasy, Michael Kerber and Ying Zheng. I would specifically thank Paul and Yuriy, who taught me how to think like a mathematician. I would like to

thank both of them, for spending many hours in front of the whiteboard and listening to my sometimes naive ideas.

I would like to thank my other dear friends as well. They are the ones who go nuts with me during the basketballs games, the ones who steal the Jello from my office, the ones who give me fake parking tickets, the ones who drive me home during my many drunken moments, and the ones who have shared five or six years of joyful graduate life with me. To list a few: Monika, Shashidhara, Thomas, Shannon, Iaryna, Jeff M., David B., Lasse, Andrew, Tony, Jeff H., Lindsay, David O., Elena, Ana Paula, Todd, Varun, Supriya, Beth, Jen, Anna, John, Hai, Guoxian, Hero, Matt, and many more (If I forget your name here, I own you a beer).

Finally I would like to thank my parents Yulin and Kay, parents-in-law Geri and John, and my big family for their support over the years. When I first started graduate school, my parents were worried that I would get sucked into the nerdy world and become a "female dinosaur" without a boyfriend. Luckily, Jeff came to the rescue.

I thank Jeff for his love.

Chapter 1

Introduction

Data and features in data. Recent advances in science and technology have created an explosion of empirical information which need data abstraction and analysis. When we only have relation data, we model them as *graphs*, which are points connected with edges, with or without geometric information. This applies to various network data, either natural, social or technological. For example, protein interaction network models protein molecules as vertices, direct-contact association and long range interactions (i.e. signal transduction) as edges. Social network models individuals or organizations as vertices, pair-wise social relations as edges. Wireless sensor networks treat sensors as vertices, transmissions between them as edges. When we have surface data, we model them as *triangulations*, which include points, edges and higher order simplicies (i.e. triangles). Triangulations encode geometric information and are convenient for computation and visualization. For example, the protein surface is modeled as a triangulation of 2-manifold in studying molecular structure. Solid modeling commonly uses triangulated representations for rendering. More generally, sometimes the object of interest is only known to us through a finite set of sample points, called *point cloud* data, from which we try to recover geometric and topological information for the original object. For example, points sam-

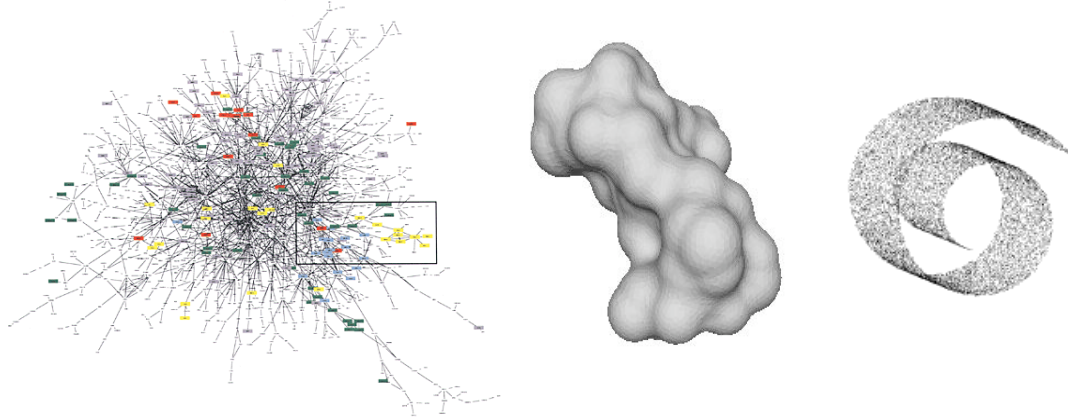


FIGURE 1.1: From left to right: a network of proteinprotein interactions in yeast with 1,548 proteins and 2,358 interactions [104]; 20,000 points sampled from the Swiss Roll data set [78]; a triangulated molecular surface.

pled from manifold, or medical data mapped as points in high dimension. Some examples are shown in Figures 1.1.

Feature extraction becomes essential to understand data that comes in various forms. First we need to define what we call features. In terms of graphs, apart from global descriptors such as degree distribution, we are interested in their clustering structure, meaning the appearance of internally densely connected groups of vertices with sparse connections between groups [87]. For protein structure, we are interested in describing the “bumps” on the surface which are protrusions and cavities important for protein docking. For point cloud data, we are interested in both global and local topological properties, such as homology and local homology.

In this thesis, we discuss techniques developed in statistics and computational topology to describe and measure features of the data. Specifically, we use spatial scan statistics inspired methods to study the structure of graphs. We adopt persistent homology to study triangulation of manifolds of dimension 1 and 2. We combine both statistical and topological methods to study homological structure of a point cloud data, and derive sampling conditions for topological inferences.

A brief history of ideas. I have always been fascinated by biological phenomena, such as protein interactions and transcription network. That is why I became interested in the *elevation function*, which was useful in identifying coarse docking configurations for protein pairs [112]. It was defined on a smoothly embedded 2-manifold in \mathbb{R}^3 and constructed based on the persistence structure of the 2-parameter family of height functions. In practice, the elevation local maximal were computed for piecewise linear triangulation of the protein surfaces which mark their cavities and protrusions. The previous algorithm computed the maxima in time $O(n^5 \log^2 n)$, where n is the number of edges in the triangulation [6]. We proposed an initial algorithm that runs in time $O(n^5)$ by dividing the Gauss sphere into regions and investigating the change of persistence pairing across regions. This initial attempt leads us to a new way of thinking, of exploring the relation between elevation and the Gaussian curvature. The resulting algorithm is described in Chapter 4. This is joint work with Dmitriy Morozov and Herbert Edelsbrunner.

At the same time I was working on the elevation function, I had an opportunity to work with the algorithm group at the HP Labs for a summer internship. The group at the time, including my collaborators Robert Schreiber, Dennis Wilkinson, Nina Mishra and Robert Tarjan, was working on graph clustering algorithms for studying web users' browsing history. Several algorithms were discussed, including the ones based on edge count and modularity from the physics literature. However, I was not quite satisfied with the clustering methods that rely on edge count but lack the statistical justification. After working with Jeff Phillips at Duke, we came up with the graph scan statistics based on hypothesis testing, which is described in Chapter 2.

With statistical tools at hand, it became rather tempting to take a statistical approach towards the persistent homology. I started focusing on a small problem, namely, the total persistence in expectation, which is discussed in Chapter 3.

My last project combines statistics with persistence. It started with the discussions with Paul Bendich and Sayan Mukherjee. Inspired by the work on local homology of stratified

spaces [20], we would like to go further as to cluster points sampled from stratified spaces into the same strata. Our proposed framework was based on mapping homology groups of small neighborhoods of pairs of points to their intersections. For the theoretical part, we first derived a topological inference theorem based on *homological feature size*. We also studied the geometric intuitions behind the topological term. Furthermore, we looked at the problem from a statistical perspective and derived a sampling condition to guarantee our inference theorems with confidence. For the algorithm part, at first, we constructed Rips complexes from the point sample and used Laplacian Eigenmaps to cluster points based on weight derived from the homology maps. We realized later that the weight matrix was ill-posed and the Rips complexes offered no topological guarantees of the underline space. We took a step back and reexamined the work by using Delaunay triangulation and Alpha complexes, this leads to the work described in Chapter 5.

Contributions. This thesis is about separating features from noise using techniques in statistics and persistent homology. We start with a statistical measure that describes features of graphs. We then study statistics on the expected total persistence, for functions defined on the triangulation of 1- and 2-manifold and general topological spaces. Subsequently, we shift our attention to the world of persistent homology alone and investigate the elevation function on the triangulation of protein surfaces. Finally, we combine both statistics and persistence to describe features of point cloud data. We now discuss the four contributions individually.

Spatial scan statistics introduced by Kulldorff [69] is commonly used to find anomalous clusters of point sets in two or higher dimension. Points in the region with the largest spatial scan statistic are most likely to be generated by a different distribution than points outside the region. In Chapter 2, We generalize spatial scan statistics from point sets to graphs and introduce a measure, the *Poisson discrepancy*, for the detection and inference of statistically anomalous clusters of a graph. We discuss the important properties of this

statistic and its relation to modularity [98] and Bregman divergences [14]. We then implement two simple greedy algorithms which seek to locally maximize the measure. Finally we illustrate the algorithm by showing its results on real-world data sets.

Continuing with the statistical scheme, in Chapter 3, we develop some theorems on total persistence. We study constant functions defined on triangulation of manifold with bounded Gaussian noise. We give upper bounds on the total persistence, in expectation, as a function of sampling parameters and properties of the triangulation.

We move to a pure persistence regime and study elevation function in Chapter 4. The elevation function on a smoothly embedded 2-manifold in \mathbb{R}^3 reflects the multiscale topography of cavities and protrusions as local maxima. The experimental study in [112] shows that using the local maxima is effective in finding initial positions during protein docking that can then be refined by local optimization. We develop a new algorithm whose worst-case running time is the same as the algorithm in [6], through its performance is roughly ten-thousand times faster for triangulated surfaces approximating smooth surfaces that we typically find in practice. We cast light on this improvement by relating the running time to the total absolute Gaussian curvature of the 2-manifold.

In Chapter 5, combining both statistics and persistence, we address the problem in *manifold learning* where the underlying space contains singularities. Specifically, we are interested in clustering points sampled from a stratified space into clusters that correspond to different components of strata. We prove a strata inference theorem such that under some geometrical and topological constraints, two points are considered similar at a given scale level. We further provide sample complexity estimates on the number of points needed to infer stratified structure with high probability. We then provide an algorithm assigning similarity measure for pairs of points and test it on synthetic data.

Chapter 2

Spatial Scan Statistics for Graph Clustering

We start the thesis by separating features from noise in graphs. By features, we mean densely connected groups of vertices. By noise, we mean the connections between these groups that interfere with the clustering process. For example, if a given graph represents a social network where its vertices are people and its edges are human relationships, then the clusters represent different social communities, and the noise come from interactions between people in different groups.

2.1 Introduction

Many networks, commonly represented as graphs, are found to exhibit modular structure, including social networks, gene regulatory network, metabolic network and the world wide web. The problem of detecting such modular structure in graphs remains outstanding. Statistical analysis has revealed some global summary statistics about these graphs, including degree distribution [15], and existence of small motifs [10]. One important approach is called *graph clustering*, that is, the detection and characterization of densely connected

groups of vertices.

Prior Work. Clustering is well-established as an important method of information extraction from large data sets. *Hard clustering* divides data into disjoint clusters while *soft clustering* allows data elements to belong to more than one cluster. Existing techniques include MCL [109], Ncut [106], graclus [46], MCODE [11], iterative scan [16], k-clique-community [97], spectral clustering [95, 86], simulated annealing [75], or partitioning using network flow [94], edge centrality [56] and functional dependencies [115].

In addition, several statistically motivated graph clustering techniques exist [66, 105, 68]. Itzkovitz et. al. discussed distributions of subgraphs in random networks with arbitrary degree sequence, which have implications for detecting network motifs [66]. Sharan et. al. introduced a probabilistic model for protein complexes taking conservation of protein sequences into consideration [105]. Koyuturk et. al. identified clusters by employing a min-cut algorithm where a subgraph was considered to be statistically significant if its size exceeded a probabilistic estimation based on a piecewise degree distribution model [68]. These techniques are all different from our approach as our model detects statistically significant clusters that are most likely to be generated by a different distribution than the baseline distribution, as this becomes clear later.

A general clustering framework using Bregman divergences as optimization functions has been proposed by Banerjee et. al. [47, 13, 14]. This approach is of note because the optimization function we use can be interpreted as a Bregman divergence, although our theoretical and algorithmic approaches are completely different.

Numerous techniques have been proposed for identifying clusters in large networks, but it has proven difficult to meaningfully and quantitatively assess them, especially from real-world data whose clustering structure is *a priori* unknown. One of the key challenges encountered by previous clustering methods is rating or evaluating the results. In large networks, manual evaluation of the results is not feasible, and previous studies have

thus turned to artificially created graphs with known structure as a test set. However, many methods, especially those in which the number of clusters must be specified in advance, give very poor results when applied to real-world graphs, which often have a highly skewed degree distribution and overlapping, complex clustering structure [56, 90].

The problem of assessment was partially solved by the introduction of *modularity* [33], a global objective function that evaluates clusters by rewarding existing internal edges and penalizing missing internal edges. Non-overlapping clusters, or partitions of a graph, are obtained by maximizing the distance from a random graph model, either by extremal optimization [48], fast greedy hierarchical algorithms [85, 97], simulated annealing [98] or spectral clustering [86].

However, modularity cannot directly assess how unexpected and thus significant individual clusters are. Additionally, it cannot distinguish between clusterings of different granularity on the same network. For example, comparable overall modularities were reported for hard clusterings of the same scientific citation graph into 44, 324, and 647 clusters [97], results which are clearly of varying usefulness depending on the application.

scan statistic [57] measure densities of data points for a sliding window on ordered data. The densest regions under a fixed size window are considered the most anomalous. This notion of a sliding window has been generalized to neighborhoods on directed graphs [96] where the neighborhood of a vertex is restricted to vertices within some constant number of edges in the graph. The number of neighbors is then compared to an expected number of neighbors based on previous data in a time-marked series.

Spatial scan statistics were introduced by Kulldorff [69] to find anomalous clusters of points in 2 or greater dimensions without fixing a window size. These statistics measure the surprise of observing a particular region by computing the log-likelihood of the most likely model for a cluster versus the probability of the most likely model for no cluster. Kulldorff argues that the region with the largest spatial scan statistic is the most likely to be generated by a different distribution, and thus is most anomalous. This test was

shown to be the most powerful test [72] for finding a region which demonstrates that the data set is not generated from a single distribution. Kulldorff [69] derived expressions for the spatial scan statistic under a Poisson and Bernoulli model. Agarwal et. al. [5] generalized this derivation to 1-parameter exponential families, and Kulldorff has studied various (see [71]) other forms of this statistic. Many techniques exist for computing the statistic quickly for anomaly detection for point sets [83, 70, 4]. Recent work by Neil et. al. [83] searched for the rectangular region with the highest density among $O(N^4)$ grids and computed its significance by randomization. They improved the naive algorithm of $O(N^4)$ to $O((N \log N)^2)$ time by partitioning the points into partially overlapping regions using a novel overlap-kd tree data structure.

Contribution. In this chapter, we generalize spatial scan statistics from point sets to graphs. The main contributions are:

- We present a measure for the detection and inference of statistically anomalous clusters of a graph. We give a measure that determines how significant the clusters are using a normalized measure of likelihood.
- We discuss some important properties of this measure and its relation to modularity and Bregman divergences.
- We implement two simple greedy algorithms which seek to locally maximize the measure. We apply these algorithms to a variety of real-world data sets, and we illustrate its ability to identify statistically significant clusters of selected granularity.

2.2 Preliminaries

In this section, we introduce the mathematical and statistical background needed to understand spatial scan statistics for graph clustering. We begin with mathematical definition of

graphs. Then we describe Poisson processes and Poisson distributions from a statistical point of view. We move on to consider Poisson random graph models which provide the foundation for our description of the graph scan statistics. We also touch base on likelihood function and hypothesis testing for non-specialists. The Poisson process definitions are from [100]. The basics on hypothesis testing are found in standard textbook [24].

Graphs. Let $G = (V, E)$ be an undirected **graph** allowing loops and multiple edges between a pair of vertices. $V = \{v_1, v_2, \dots, v_{|V|}\}$ is the vertex set where $|V|$ is its size. $[V]^2 = \{\{v_i, v_j\} | v_i, v_j \in V\}$ is the set of 2-element multisets called *edges*, that is, the two endpoints of an edge are not necessarily different. E is a multiset of edges in $[V]^2$ and $c(V) = |E|$ is its size. Let $c_0(x)$ be the multiplicity of the edge $x \in [V]^2$ in E . $d = \{d_1, d_2, \dots, d_{|V|}\}$ is the degree sequence for V , where d_i is the *degree* of a vertex v_i , that is, the number of edges in E that contain v_i . A loop at v_i is counted twice in the degree. $d_U = \sum_{v_i \in U} d_i$ is the *total degree* of a subset U of the vertex set.

A *cluster* is a subset of vertices $W \subseteq V$. W induces a subgraph $G(W) = (W, E(W))$ with $E(W)$ containing all edges $x = \{v_i, v_j\} \in E$ with $v_i, v_j \in W$. The collection of all clusters $W \subseteq V$ is denoted as \mathcal{W} . Define $c(W)$ as the number of edges in $G(W)$, $c(W) = |E(W)| = \sum_{x \in [W]^2} c_0(x)$.

$sd(U, W)$ is the size of the symmetric difference between $U, W \subseteq V$, $sd(U, W) = |U| + |W| - 2|U \cap W|$. $Lk(U)$ is the *link* of a vertex set $U \subseteq V$ defined as $Lk(U) = \{v_j \in V \setminus U \mid \{v_i, v_j\} \in E \text{ and } v_i \in U\}$.

Poisson process and Poisson distribution. The *Poisson process* is a counting process with a set of random variables $\{N(t), t \geq 0\}$, where $N(t)$ counts the number of events occur up to time t , for $0 \leq t < \infty$. It has the following properties [100]:

- $N(0) = 0$.

- (Independent increments) $N(t) - N(s)$ and $N(u) - N(v)$ are independent for non-overlapping intervals $(s, t]$ and $(v, u]$.
- (Stationary increments) The distribution of $N(t) - N(s)$ only depends on the length of the interval $(s, t]$.
- No counted occurrences are simultaneous.

As a consequence of the above definition, the probability distribution of $N(t)$ is a *Poisson distribution* with parameter λ . Here, λ is the expected number of events that occur during an interval of length t . Let X be a random variable denoting the number of events that occur during that interval, that is, $X = N(t + s) - N(s) = N(t) - N(0) = N(t)$. Then

$$\Pr(X = k) = \frac{e^{-\lambda} \lambda^k}{k!}.$$

We say X is distributed as a *Poisson random variable* with *intensity* λ , $X \sim \text{Poi}(\lambda)$.

If a random variable X is Poisson distributed with parameter λ , it has the following basic properties.

- (1) $\sum_{k \neq 0} \Pr(X = k) = 1$.
- (2) X has mean λ .
- (3) X has variance λ and standard deviation $\sqrt{\lambda}$.
- (4) Implied by (3), if a random selection is made from a Poisson process with intensity λ such that each event is selected with probability p , independently of the others, the resulting process is a Poisson process with intensity $p\lambda$ [111]. p is referred to as the *random selection rate*.
- (5) If $X_i \sim \text{Poi}(\lambda_i), i = 1, \dots, N$ where X_i are independent, then $Y = \sum_{i=1}^N X_i \sim \text{Poi}(\sum_{i=1}^N \lambda_i)$.

- (6) The Poisson distribution is the limit of a binomial distribution for which the number of trials, n , approaches infinity and the probability of success on each trial, p , approaches 0 in such a way that $\lambda = np$ [99]. The details are shown below.

We have

$$\Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}.$$

Setting $\lambda = np$,

$$\begin{aligned} \Pr(X = k) &= \frac{n!}{k!(n-k)!} \left(\frac{\lambda}{n}\right)^k \left(1 - \frac{\lambda}{n}\right)^{n-k} \\ &= \frac{\lambda^k}{k!} \frac{n!}{(n-k)!n^k} \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-k}. \end{aligned}$$

As $n \rightarrow \infty$, $\frac{n!}{(n-k)!n^k} \rightarrow 1$, $\left(1 - \frac{\lambda}{n}\right)^n \rightarrow e^{-\lambda}$, $\left(1 - \frac{\lambda}{n}\right)^{-k} \rightarrow 1$. Substituting the limits in the expression for the binomial distribution gives the Poisson frequency function, $\Pr(X = k) = e^{-\lambda} \lambda^k / k!$.

For the rest of this chapter, we say a random variable X is Poisson distributed, that is, $X \sim \text{Poi}(\lambda)$, if its distribution is approximated by the Poisson distribution with λ in the limit as n goes to ∞ .

Poisson random graph model. Many large real-world graphs have diverse and non-uniform degree distributions [15, 8, 7] that are not accurately described by the classic Erdős and Rényi random graph models [53]. We consider a Poisson random graph model here that captures some main characteristics of real-world graphs, specifically, allowing vertices to have different expected degrees. Notice that this model is different from models used in [30, 33].

We are given a graph $G = (V, E)$ with a vertex set V degree sequence d . The total degree of vertices in G is $d_V = \sum_{v_i \in V} d_i$. The number of edges in G is $m = c(V) =$

$$|E| = d_V/2.$$

We first describe a *binomial random graph model* which generates random graphs with the above two parameters V and d . In the limit, it becomes the *Poisson random graph model*. It chooses a total of m pairs of vertices as edges through m steps, with replacement. At each step, each of the two end-points of an edge is chosen among all vertices with probability proportional to their degrees.

A random graph generated under the model has the following properties:

- (1) The number of edges is m (by definition).
- (2) The expected degree of vertex v_i is d_i (by construction), $m(d_i/2m + d_i/2m) = d_i$.
- (3) The probability that a single draw generates the edge $x = \{v_i, v_j\}$ is

$$p(x) = \begin{cases} d_i d_j / 2m^2 & i \neq j \\ d_i^2 / 4m^2 & i = j \end{cases}$$

- (4) The expected multiplicity of the edge $x = \{v_i, v_j\}$ after m draws is therefore

$$\mu_0(x) = mp(x) = \begin{cases} d_i d_j / 2m & i \neq j \\ d_i^2 / 4m & i = j \end{cases}$$

- (5) For a cluster $W \subseteq V$, the expected number of edges connecting vertex pairs in $[W]^2$ is,

$$\mu(W) = \sum_{x \in [W]^2} \mu_0(x) = d_W^2 / 4m.$$

Note that when $W = V$, we get $\mu(V) = d_V^2 / 4m = m$, which is consistent with (1).

It is important to note that the binomial random graph model chooses exactly m edges. Let $\mathcal{N}(x)$ be a random variable describing the multiplicity of the edge $x = \{v_i, v_j\}$. By construction, $\Pr(X = k) = \binom{m}{k} p(x)^k (1 - p(x))^{m-k}$. As shown previously, in the limit,

as $m \rightarrow \infty$, $p(x) \rightarrow 0$ such that $\mu_0(x) = mp(x)$, $\mathcal{N}(x)$ becomes Poisson distributed with parameter $\mu_0(x)$.

We now introduce the *Poisson random graph model* which is equivalent to the limiting behavior of the Binomial random graph model. It generated random graph with *graph Poisson process*. For the rest of the chapter, we use this model to derive our spatial scan statistics. The Poisson random graph model chooses m edges in expectation. More precisely, for each pair of vertices x , the number of edges between them is drawn from a Poisson distribution of parameter $\mu_0(x)$. Therefore, a graph generated by the Poisson random graph model has the following properties,

- (1) The expected number of edges is m .
- (2) The expected degree of vertex i is $\sum_{j \neq i} d_i d_j / 2m + 2d_i^2 / 4m = d_i$, given $\sum_{j \neq i} d_j = 2m - d_i$.
- (3) By definition, let $\mathcal{N}(x)$ be a random variable describing the multiplicity of the edge $x = \{v_i, v_j\}$,

$$\mathcal{N}(x) \sim \text{Poi}(\mu_0(x)).$$

The expected value of $\mathcal{N}(x)$ is $\mu_0(x)$.

- (4) Let $\mathcal{N}(W)$ be a random variable describing the number of edges connecting vertex pairs in $[W]^2$, $\mathcal{N}(W) = \sum_{x \in [W]^2} \mathcal{N}(x)$ and,

$$\mathcal{N}(W) \sim \text{Poi}(\mu(W)).$$

Note that when $W = V$, we get $\mathcal{N}(V) \sim \text{Poi}(\mu(V)) = \text{Poi}(m)$, that is, the expected number of edges in the graph is m , consistent with (1).

Correspondingly, if a random selection rate p is associated with a graph Poisson process of intensity λ , then we have a graph Poisson process with intensity $p\lambda$.

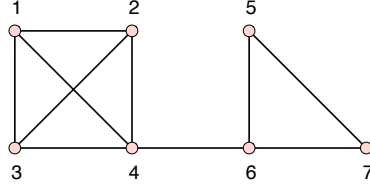


FIGURE 2.1: Example of a cluster and its various measures.

A simple example is shown in Figure 2.1, where cluster $W = \{v_1, v_2, v_3, v_4\}$. The total number of edges is $c(V) = 10$. The number of edges in the cluster W is $c(W) = 6$. The number of edges between vertex pairs are $c_0(\{v_1, v_1\}) = 0$, $c_0(\{v_1, v_2\}) = 1$. The total number of expected edges is $\mu(V) = 10$. The number of expected edges in W is $\mu(W) = 169/40$. The number of expected edges between vertex pairs are $\mu_0(\{v_1, v_1\}) = 3 \times 3 / (4 \times 10) = 9/40$, $\mu_0(\{v_1, v_2\}) = 3 \times 3 / (2 \times 10) = 9/20$. Notice that in this example $c(W) \neq \mu(W)$.

Bipartite extensions. Many data sets which are applicable to the Poisson random graph model are bipartite graphs. Thus we derive here the bipartite extensions to our definitions.

An undirected graph $G = (V, E)$ is *bipartite* if there is a partition $V = X \cup Y$, with X and Y disjoint, and E is a multiset of edges in $[XY]$, where $[XY] = \{\{v_i, v_j\} | v_i \in X, v_j \in Y\}$. An element $x = \{v_i, v_j\} \in [XY]$ is defined as an *edge*. G does not allow loops but allows multiple edges between a pair of distinct vertices. A cluster is a subset of vertices $W \subseteq V$, where $W = X_W \cup Y_W$, $X_W \subseteq X$ and $Y_W \subseteq Y$. $[X_W Y_W] = \{\{v_i, v_j\} | v_i \in X_W, v_j \in Y_W\}$. W induces a subgraph $G(W) = (W, E(W))$.

Let m be the total number of edges in the bipartite graph, $m = \sum_{v_i \in X} d_i = \sum_{v_j \in Y} d_j$. The *bipartite binomial random graph model* differs from the previous model that it does not allow edges between vertices in the same partition. A graph generated by the model has the following properties:

- (1) The number of edges is m (by definition).

(2) The expected degree of vertex v_i is d_i (by construction), that is, $m(d_i/m) = d_i$.

(3) The probability that we draw the edge $x = \{v_i, v_j\}$, $v_i \in X$, $v_j \in Y$ is

$$p(x) = d_i d_j / m^2.$$

(4) The expected multiplicity of the edge $x = \{v_i, v_j\}$ is

$$\mu_0(x) = mp(x) = d_i d_j / m.$$

(5) For a cluster $W \subseteq V$, the expected number of edges connecting vertex pairs in $[X_W Y_W]$ is

$$\mu(W) = \sum_{x \in [X_W Y_W]} \mu_0(x) = d_X d_Y / m.$$

Notice that for $W = V$, this gives $\mu(V) = m$.

We derive the *bipartite Poisson random graph model* similarly as the limit of the binomial model.

Likelihood function and maximum likelihood estimator. In statistical inference, a likelihood function (or likelihood) is a function of the parameters of a statistical model that allows estimation of unknown parameters based on known outcomes. For a discussion, see [17]. It is important to understand the distinction between “likelihood” and “probability” as the latter is used to predict unknown outcomes based on known parameters. Formally, let $f(x|\theta)$ denote the probability density function of the random variable X . That is, over any range R , $\Pr(X \in R) = \int_{x \in R} f(x|\theta) dx$, based on a known parameter θ . Then, the function of θ defined by

$$L(\theta|x) = f(x|\theta)$$

is call the *likelihood function* ([24], page 290). $L(\theta|x)$ is viewed as a function of θ with x fixed, while $f(x|\theta)$ is viewed as a function of x with θ fixed. Intuitively, see Figure 2.2.

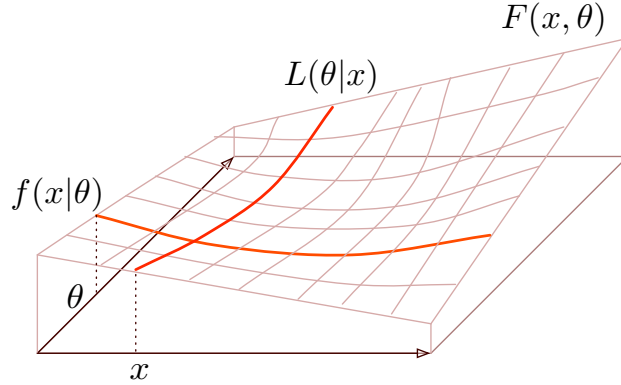


FIGURE 2.2: Intuitive understanding of likelihood and probability.

We are given a two parameter function $F(x, \theta)$, if we fix θ , we obtain probability density function $f(x|\theta)$; if we fix x , we obtain likelihood function $L(\theta|x)$. Both are visualized as “slices” of $F(x, \theta)$. When both slices intersect, that is when $L(\theta|x) = f(x|\theta)$.

It is important to note that $L(\theta|x)$ does not imply a conditional on x , to be consistent with literature, $L(\theta|x)$ is sometimes denoted as $L(\theta; x)$. However, here we adapt the former notation used in [24].

Another concept closely related to the likelihood function is the maximum likelihood estimator defined below.

Definition 2.2.1 (Definition 7.2.4 in [24]). *For sample point x , let $\hat{\theta}(x)$ be a parameter value at which $L(\theta|x)$ obtains its maximum as a function of θ with x fixed. A **maximum likelihood estimator (MLE)** of the parameter θ based on a sample X is $\hat{\theta}(X)$.*

Hypothesis testing and likelihood ratio test. For non-specialist, we review an important frequentists’ inference method, called hypothesis testing. The rest of this section is standard textbook definitions from [24].

Definition 2.2.2 (Definition 8.1.1, 8.1.2 and 8.1.3 in [24]). *A **hypothesis** is a statement about a population parameter. The two complementary hypotheses in hypothesis testing are called the **null hypothesis** and the **alternative hypothesis**, denoted by H_0 and H_1 , re-*

spectively. A **hypothesis testing** is a rule that specifies: 1) For which sample values the decision is made to accept H_0 as true; 2) For which sample values H_0 is rejected and H_1 is accepted as true.

If θ denotes a population parameter, the general format of the null and alternative hypothesis is $H_0 : \theta \in \Theta_0, H_1 : \theta \in \Theta_0^c$, where Θ_0 is some subset of the parameter space Θ and Θ_0^c is its complement ([24], page 373). The subset of the sample space for which H_0 is rejected is called the **rejection region** or **critical region**. A hypothesis test is usually specified in terms of a **test statistic**, which is a function of the sample. The **likelihood ratio test statistic** for testing $H_0 : \theta \in \Theta_0$ versus $H_1 : \theta \in \Theta_0^c$ is defined as follows, which equals the ratio of the maximum likelihood values,

$$\lambda(x) = \frac{\sup_{\Theta_0} L(\theta|x)}{\sup_{\Theta} L(\theta|x)} = \frac{L(\hat{\theta}_0|x)}{L(\hat{\theta}|x)}.$$

A **likelihood ratio test** (LRT) is any test that has a critical region of the form $\{x : \lambda(x) \leq c\}$ for some $0 \leq c \leq 1$. It has the following properties [107],

- $0 \leq \lambda(x) \leq 1$;
- When $\hat{\theta}$ is far away from $\hat{\theta}_0$, $\lambda(x)$ is small, then H_0 should be rejected.

It is important to note that sometimes the reciprocal is used as the definition, as we will see in the case of spatial scan statistic. The alternative definition of the LRT statistic is,

$$\lambda(x) = \frac{L(\hat{\theta}|x)}{L(\hat{\theta}_0|x)},$$

where $\lambda(x) \geq 1$, when $\lambda(x)$ is large, H_0 should be rejected.

Uniformly most powerful test. A hypothesis test might make one of two types of errors. Let R be the rejection region for a test. If $\theta \in \Theta_0$ but the test incorrectly decides to reject H_0 , it makes the **Type I Error**, the probability of making Type I error is denoted as

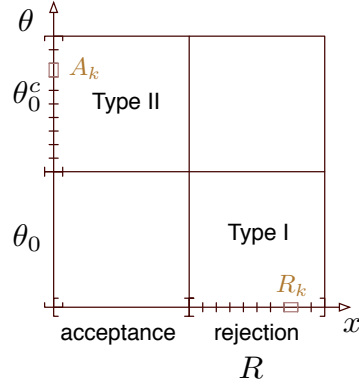


FIGURE 2.3: Type I and Type II error.

$P_\theta(X \in R)$, which is commonly known as the *false positive rate*. If $\theta \in \Theta_0^c$ but the test incorrectly decides to accept H_0 , it makes the *Type II Error*, the probability of making Type II error is denoted as $P_\theta(X \in R^c) = 1 - P_\theta(X \in R)$ [24], which is commonly known as the *false negative rate*. Both Type I and Type II error are shown in Figure 2.3.

The *power function* of a hypothesis test with rejection region R is the function of θ defined by $\beta(\theta) = P_\theta(X \in R)$ [24]. For $\theta \in \Theta_0$, it equals the probability of a Type I Error. For $\theta \in \Theta_0^c$, it equals one minus the probability of a Type II Error. The *power* of a test is the probability that the test will not make Type II error. In practice, a good test has power function near 0 for most $\theta \in \Theta_0$ and near 1 for most $\theta \in \Theta_0^c$ [24].

For $0 \leq \alpha \leq 1$, a test with power function $\beta(\theta)$ is a *level α test* if $\sup_{\theta \in \Theta_0} \beta(\theta) \leq \alpha$ [24]. α is called the *significance level*.

Definition 2.2.3 (Definition 8.3.11 in [24]). *Let \mathcal{C} be a class of tests for testing $H_0 : \theta \in \Theta_0$ versus $H_1 : \theta \in \Theta_0^c$. A test in class \mathcal{C} with power function $\beta(\theta)$, is a **uniformly most powerful (UMP) class \mathcal{C} test** if $\beta(\theta) \geq \beta'(\theta)$ for every $\theta \in \Theta_0^c$ and every $\beta'(\theta)$ that is a power function of a test in class \mathcal{C} .*

Simply put, a test is UMP if it has a smaller Type II error than all other tests in the same class. In short, level α test controls the probability of a Type I Error while UMP controls that of a Type II Error.

Individually most powerful test. As noted by Kulldorff, we cannot expect to find a UMP test except for the special case when there is only one cluster in the alternative hypothesis. He then defines *individually most powerful (IMP) test* as follows [69]. Intuitively, when a test is compared to any other test with the same Type I error and same rejection region R except for a subset $R_k \subset R$, it is IMP if it has a smaller Type II error in R_k .

The parameter space Θ_0^c is partitioned into a countable number of subsets $\{A_j\}$. Likewise using the same index, the critical region R is partitioned into subsets $\{R_j\}$. Let R' denote an alternative critical region with corresponding disjoint subsets, $\{R'_j\}$.

Definition 2.2.4. A test is **individually most powerful** with respect to a partition $\{A_j\}$ of the parameter space Θ_0^c , and a partition $\{R_j\}$ of the critical region R , if for each A_k there are no sets R' and $\{R'_j\}$ such that: assuming a significance level α ,

- $\beta(\theta) = \beta'(\theta)$ if $\theta \notin A_k$.
- $\beta(\theta) < \beta'(\theta)$ if $\theta \in A_k$.

This is illustrated in Figure 2.3.

This means, if we fix the critical region except for its subset R_k , then the test is uniformly most powerful compared to all remaining choices of the critical region and with respect to all parameters $\theta \in A_k$.

2.3 Graph Scan Statistic

In this section we generalize the notion of a spatial scan statistic [69] from point sets to graphs, we call our measure the *graph scan statistic*. We highlight some important properties of this statistic, as well as its relation to local modularity and Bregman divergences. It is this graph scan statistic that we use to provide quantitative assessment of significant clusters.

Overview. We are given a graph $G = (V, E)$ with vertex set V and degree sequence d . Here we consider two graph Poisson processes based on these two parameters, V and d .

For a fixed cluster $W \subseteq V$, a graph Poisson process generates a random graph with parameters V , d and two random selection rates p and q , such that edges in the graph are generated with rate p and edges outside are generated with rate q . The first process requires that $p = q$ and generates a random graph $G_0 = (V, E_0)$. Let X_0 be the random variable describing the number of edges in G_0 , $X_0 \sim \text{Poi}(p\mu(V))$. The second process requires $p > q$ and generates a random graph $G_1 = (V, E_1)$. Let X_1 be the random variable describing the number of edges in G_1 , $X_1 \sim \text{Poi}(p\mu(W) + q(\mu(V) - \mu(W)))$.

The graph scan statistic is a likelihood ratio test statistic. The null hypothesis H_0 assumes that $G = G_0$. That is, all edges in G are generated at the same rate. The alternative hypothesis H_1 assumes that $G = G_1$. That is, the edges in the cluster are generated at a higher rate than those outside. Since we are interested in detecting dense clusters, we ignore the part of parameter space where $p < q$.

Our null hypothesis H_0 and alternative H_1 hypothesis are,

- $H_0 : p = q$.
- $H_1 : p > q$.

For a fixed W , the likelihood function under H_0 and H_1 are $L_0 = L(p, q|G = G_0)$ and $L_1 = L(p, q|G = G_1)$, respectively. L_0 obtains its maximum with its maximum likelihood estimators \hat{p}, \hat{q} , let $\hat{L}_0 = L(\hat{p}, \hat{q}|G = G_0)$. Similarly, let $\hat{L}_1 = L(\hat{p}, \hat{q}|G = G_1)$. We obtain the maximum likelihood function $\hat{L} = L(p, q|G)$ over the entire parameter space as,

$$\hat{L} = \begin{cases} \hat{L}_1 & \text{if } \hat{p} > \hat{q}, \\ \hat{L}_0 & \text{otherwise.} \end{cases}$$

The likelihood ratio test statistic for a fixed W is

$$\lambda = \frac{\hat{L}}{\hat{L}_0} = \begin{cases} \frac{\hat{L}_1}{\hat{L}_0} & \text{if } \hat{p} > \hat{q}, \\ 1 & \text{otherwise.} \end{cases}$$

As we vary W , the LRT statistic is a function of W . Specifically, \hat{L} is a function of W . That is, $\lambda = \lambda(W) = \frac{\hat{L}(W)}{\hat{L}_0}$. Our graph scan statistic is defined as,

$$\Lambda = \max_{W \in \mathcal{W}} \lambda(W).$$

Derivation. We start our derivation with $\lambda(W)$ for a fixed W . Under H_0 , since $p = q$, we compute $L_0 = L(p, q|G = G_0) = L(p|G = G_0)$. The probability of $c(V)$ edges being observed in G_0 is $\Pr(X_0 = c(V))$, where $X_0 \sim \text{Poi}(p\mu(V))$. The probability that a single draw generates the edge x is $f(x) = \frac{\mu_0(x)}{\mu(V)}$. Let ξ be the number of orderings of edges in G_0 (to be exact, the number of permutations of a multiset). Therefore,

$$\begin{aligned} L_0 &= \Pr(X_0 = c(V)) \xi \prod_{x \in G_0} f(x) \\ &= \frac{e^{-p\mu(V)} (p\mu(V))^{c(V)}}{c(V)!} \xi \prod_{x \in G_0} \frac{\mu_0(x)}{\mu(V)} \\ &= \frac{e^{-p\mu(V)} p^{c(V)}}{c(V)!} \xi \prod_{x \in G_0} \mu_0(x). \end{aligned}$$

We compute a MLE of p as $\hat{p} = \frac{c(V)}{\mu(V)} = 1$. The maximized L_0 is

$$\hat{L}_0 = \frac{e^{-c(V)}}{c(V)!} \xi \prod_{x \in G_0} \mu_0(x).$$

Under H_1 , we compute $L_1 = L(p, q|G = G_1)$. The probability of $c(V)$ edges being observed in G_1 is $\Pr(X_1 = c(V))$, where $X_1 \sim \text{Poi}(p\mu(W) + q(\mu(V) - \mu(W)))$. The probability that a single draw generates the edge x is,

$$f(x) = \begin{cases} \frac{p\mu_0(x)}{p\mu(W) + q(\mu(V) - \mu(W))} & x \in E_1(W), \\ \frac{q\mu_0(x)}{p\mu(W) + q(\mu(V) - \mu(W))} & x \in E_1 \setminus E_1(W). \end{cases}$$

Therefore, for a fixed W ,

$$\begin{aligned}
L_1(W) &= Pr(X_1 = c(V)) \xi \prod_{x \in E_1(W)} f(x) \prod_{x \in E_1 \setminus E_1(W)} f(x) \\
&= \frac{e^{-p\mu(W) - q(\mu(V) - \mu(W))}}{c(V)!} (p\mu(W) + q(\mu(V) - \mu(W)))^{c(V)} \\
&\quad \cdot \xi \prod_{x \in E_1(W)} \frac{p\mu_0(x)}{p\mu(W) + q(\mu(V) - \mu(W))} \prod_{x \in E_1 \setminus E_1(W)} \frac{q\mu_0(x)}{p\mu(W) + q(\mu(V) - \mu(W))} \\
&= \frac{e^{-p\mu(W) - q(\mu(V) - \mu(W))}}{c(V)!} p^{c(W)} q^{c(V) - c(W)} \xi \prod_{x \in E_1} \mu_0(x).
\end{aligned}$$

For a fixed W , L_1 takes its maximum at $\hat{p} = \frac{c(W)}{\mu(W)}$, $\hat{q} = \frac{c(V) - c(W)}{\mu(V) - \mu(W)}$, and

$$\hat{L}_1(W) = \frac{e^{-c(V)}}{c(V)!} \left(\frac{c(W)}{\mu(W)} \right)^{c(W)} \left(\frac{c(V) - c(W)}{\mu(V) - \mu(W)} \right)^{c(V) - c(W)} \xi \prod_{x \in E} \mu_0(x).$$

The likelihood function of the entire parameter space with its MLEs is,

$$\hat{L}(W) = \begin{cases} \hat{L}_1(W) & \text{if } \frac{c(W)}{\mu(W)} > \frac{c(V) - c(W)}{\mu(V) - \mu(W)}, \\ \hat{L}_0 & \text{otherwise.} \end{cases}$$

Given $W \in \mathcal{W}$, we define the *likelihood ratio* $\lambda(W)$ as

$$\lambda(W) = \frac{\hat{L}(W)}{\hat{L}_0}.$$

The *graph scan statistic*, Λ , is the maximum likelihood ratio over all clusters $W \in \mathcal{W}$,

$$\Lambda = \max_{W \in \mathcal{W}} \lambda(W).$$

If there is at least one cluster $W \in \mathcal{W}$ such that $\frac{c(W)}{\mu(W)} > \frac{c(V) - c(W)}{\mu(V) - \mu(W)}$, we define

$$\Lambda = \max_{W \in \mathcal{W}} \left(\frac{c(W)}{\mu(W)} \right)^{c(W)} \left(\frac{c(V) - c(W)}{\mu(V) - \mu(W)} \right)^{c(V) - c(W)},$$

otherwise, $\Lambda = 1$.

Simplification. Let $r(W) = \frac{c(W)}{c(V)}$ and $b(W) = \frac{\mu(W)}{\mu(V)}$. Notice that $c(V) = \mu(V)$. We define the *Poisson discrepancy*, d_P , as

$$d_P(W) = r(W) \log \frac{r(W)}{b(W)} + (1 - r(W)) \log \frac{1 - r(W)}{1 - b(W)}.$$

Intuitively, $r(W)$ is the observed edge ratio and $b(W)$ is the baseline edge ratio in $G(W)$ and G .

Since $\log \Lambda = c(V) \max_{W \in \mathcal{W}} d_P(W) = \max_{W \in \mathcal{W}} \log \lambda(W)$, for the cluster W that maximizes d_P , $d_P(W)$ constitutes the graph scan statistic Λ . This means that the likelihood test based on $\max_{W \in \mathcal{W}} d_P(W)$ is identical to one based on Λ .

Since $0 < r(W), b(W) \leq 1$, from this point on, for computational purpose, we evaluate clusters based on the Poisson discrepancy. d_P determines how surprising $r(W)$ is compared to the rest of the distribution. Thus clusters with larger values of d_P are more likely to be inherently different from the rest of the data.

Significance of a Cluster. Although we can consider all clusters $W \in \mathcal{W}$ and determine the one that is most anomalous by calculating the Poisson discrepancy, this does not determine whether this value is significant. Even a graph generated according to a Poisson random graph model will have some cluster which is most anomalous. For a graph $G = (V, E)$ with degree sequence d and for a particular cluster W we can compare $d_P(W)$ to the distribution of the values of the most anomalous clusters found in a large set of (say 1000) random data graphs. To create a random data graph, we fix V ; then we randomly select $c(V)$ edges according to a Poisson random graph model with expected degree sequence d . If the Poisson discrepancy for the original cluster is greater than all but a α fraction of the most anomalous clusters from the random data sets, then we say it has a *p-value* of α , i.e. $\alpha = 0.05$. The lower the *p-value*, the more significantly anomalous the range is. These high discrepancy clusters are most significant because they are the most unlikely compared to what is expected from our random graph model.

2.3.1 Properties of Graph Scan Statistics

Kulldorff has proved some optimal properties for the likelihood ratio test statistic for point sets [69, 72]. In the context of graphs, we describe those properties essential for detecting statistically anomalous clusters in terms of d_P . For details and proofs, see Appendix A. As direct consequences of Theorem 1 and 2 in [69], we have

Theorem 2.3.1. *Let $\mathcal{X} = \{x_i | x_i \in E\}_{i=1}^{c(V)}$ be the set of edges in $G = (V, E)$ where \hat{W} is the most likely cluster. Let $\mathcal{X}' = \{x'_i | x'_i \in [V]^2\}_{i=1}^{c(V)}$ be an alternative configuration of a graph $G' = (V, E')$ where $\forall x_i \in E(\hat{W}), x'_i = x_i$. If the null hypothesis is rejected under \mathcal{X} , then it is also rejected under \mathcal{X}' .*

Intuitively, as long as the edges within the subgraph constituting the most likely cluster are fixed, the null hypothesis is rejected no matter how the rest of the edges are shuffled around.

This theorem implies that:

1. $d_P(W)$ does not change as long as its internal structure and the total number of reported edges outside W remains the same. Intuitively, clusters defined by other vertex subsets do not affect the discrepancy on W . Formally, $d_P(W)$ is independent of the value of $c_0(x)$ for any edge $x \in E \setminus E(W)$, as long as $c(V) - c(W)$ remains unchanged.
2. If the null hypothesis is rejected by d_P , then we can identify a specific cluster that is significant and implies this rejection. This distinguishes between saying “there exist significant clusters” and “the cluster W is a significant cluster,” where d_P can do the latter.

Theorem 2.3.2. *d_P is individually most powerful for finding a single significant cluster: for a fixed false positive rate and for a given set of subgraphs tested, it is more likely to detect over-density than any other test statistic [83].*

This is paramount for effective cluster detection. It implies that:

3. We can determine the single edge $x \in [V]^2$ (or set of edges) that will most increase the Poisson discrepancy of a cluster, and thus most decrease its p -value.

2.3.2 Graph Scan Statistics and Local Modularity

Several local versions of modularity have been used to discover local community structure [32, 81]. Specifically, local modularity introduced in [98] is used to find the community structure around a given node. The *local modularity* of $W \subseteq V$ measures the difference between the number of observed edges among vertex pairs in $[W]^2$ and the number expected, $\mu(W)$,

$$Q_\gamma(W) = c(W) - \gamma\mu(W).$$

One approach to clustering is to find the cluster W that locally maximizes Q_γ . The γ parameter with default value 1, is a user specified knob [98] that scales the expected number of edges among vertex pairs in $[W]^2$ under a Poisson random graph model. We observe that it effectively tunes the size of the clusters which optimize $Q_\gamma(W)$. For a fixed cluster W , Q_γ can be treated as a linear function of γ , where its intersection with the Y -axis is $c(W)$, and its slope is $-\mu(W)$. Q_γ for all $W \in \mathcal{W}$ forms a family of linear functions whose upper envelope corresponds to clusters that maximize Q as γ varies. It can be observed that as γ increases, $c(W)$ is non-increasing and $\mu(W)$ is non-decreasing for the cluster W that maximizes Q_γ . When $\gamma = 1$, we denote $Q = Q_1$.

It is important to distinguish Q from d_P . While Q measures the edge distance from the expected random graph, d_P measures the difference in how likely the total number of edges are to occur in a general random graph and how likely they are to occur in cluster W and its complement as separate random graph models. To summarize, Q calculates the distance, and spatial scan statistics measure how unexpected this distance is, given W .

Using some machinery developed by Agarwal et. al. [5] we can create an ε -approximation

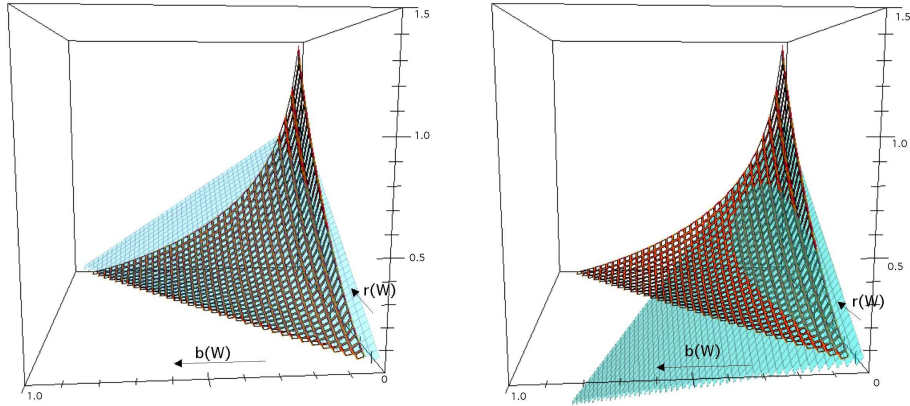


FIGURE 2.4: Comparison of d_P (gridded) to $\frac{1}{m}Q_1$ (transparent, left panel) and $\frac{1}{m}Q_2$ (transparent, right panel) over $(r(W), b(W)) \in [0, 1]^2$ such that $r(W) > b(W)$. Recall that $r(W)$ and $b(W)$ are the actual and expected fraction of a graph's edges which lie in a particular cluster; for applications to large networks, a range of say $(0, 0.2)^2$ is most pertinent to clustering. For this range, Q_2 is shown to approximate d_P more closely than Q_1 .

of d_P with $O(\frac{1}{\varepsilon} \log^2 |V|)$ linear functions with parameters $r(V)$ and $b(V)$, in the sense that the upper envelope of this set of linear functions will be within ε of d_P . We can interpret Q_γ as a linear function whose slope is controlled by the value of γ . Figure 2.4 shows how Q_1 and Q_2 , respectively, approximate d_P . Thus we can find the optimal cluster for $O(\frac{1}{\varepsilon} \log^2 |V|)$ values of γ and let W be the corresponding cluster from this set which has the largest value of $d_P(W)$. Let \hat{W} be the cluster that has the largest value of $d_P(\hat{W})$ among all possible clusters. Then $d_P(W) + \varepsilon \geq d_P(\hat{W})$.

However, a further study of Agarwal et. al. [4] showed that a single linear function (which would be equivalent to $\gamma = 2$ for Q_γ) approximated d_P on average to within about 95% for a problem using point sets. Note in Figure 2.4 how Q_2 seems to approximate d_P better than Q_1 , at least for a large portion of the domain containing smaller clusters.

Heuristic measure $d_{P,\gamma}$. Several properties are also shared between d_P and Q . The tuning knob γ can be used in Poisson discrepancy to scale the expected number of edges

among vertex pairs in $[W]^2$.

$$d_{P,\gamma}(W) = r(W) \log \frac{r(W)}{\gamma b(W)} + (1 - r(W)) \log \left(\frac{1 - r(W)}{1 - b(W)} \right)$$

Technically, the function $d_{P,\gamma}$ describes the effect of scaling by γ the expected number of edges among vertex pairs in $[W]^2$ (but not outside the cluster), while not allowing q , the parameter to model the random graph outside this cluster, to reflect γ . Thus in the same way as with Q_γ for large γ , clusters need to have significantly more edges than expected to have a positive d_P value. The following lemma highlights this relationship.

Lemma 2.3.3. *Consider two clusters W_1 and W_2 such that $d_{P,\gamma}(W_1) = d_{P,\gamma}(W_2)$ and that $c(W_1) > c(W_2)$. Then for any $\delta > 0$ we know $d_{P,\gamma+\delta}(W_1) < d_{P,\gamma+\delta}(W_2)$.*

The same property holds for Q_γ and μ in place of $d_{P,\gamma}$ and c , respectively. That is, consider two clusters W_1 and W_2 such that $Q_\gamma(W_1) = Q_\gamma(W_2)$ and that $\mu(W_1) > \mu(W_2)$. Then for any $\delta > 0$ we know $Q_{\gamma+\delta}(W_1) < Q_{\gamma+\delta}(W_2)$.

Proof. We can write

$$d_{P,\gamma}(W) = d_P(W) - r(W) \log \gamma,$$

thus as γ increases $d_{P,\gamma}(W_1)$ will decrease faster than $d_{P,\gamma}(W_2)$.

We can also write

$$Q_\gamma(W) = c(V)(r(W) - \gamma b(W)) = Q_1(W) - c(V)(\gamma - 1)b(W).$$

Thus the same argument applies. □

This implies that for the discrepancy measure, we should expect the size of the optimal clusters to be smaller as we increase γ , as is empirically demonstrated in Section 2.5.

2.3.3 Graph Scan Statistics and Bregman Divergences

Many Bregman divergences, including the KL-divergence, can be interpreted as spatial scan statistics. The KL-divergence is a measure of the difference between two probability

distributions $\alpha, \beta \in \mathbb{R}^d$ such that $D_{KL}(\alpha, \beta) = \sum_{i=1}^d \alpha_i \log \alpha_i / \beta_i$. The KL-divergence between two 2-point distributions is equivalent to d_P up to a constant factor.

Banerjee et. al. use Bregman divergences in a different way than does this chapter. In the context of graph bi-clustering, Bregman hard clustering finds a bi-partitioning and a representative for each of the partitions such that the expected Bregman divergence of the data points (edges) from their representatives is minimized. For details and derivations, see [14].

The basic idea is as follows. Given a graph $G = (V, E)$ and a random cluster W . W induces a bipartition of edges in G . We compute a “center” edge for each partition. Then we reassign each edge to its “closest” center where the closeness is measured by KL-divergence. We iterate this process until the centers converge. Therefore we obtain a bipartition of the graph.

More precisely, given a graph $G = (V, E)$, let n be the number of potential edges in G , $n = c(V)^2$. Set $[V]^2 = \{x_1, x_2, \dots, x_n\}$ has probability measure μ_0 . We start with a random cluster $W \subseteq V$ which induces a bi-partitioning of edges in G . That is, for each $x_i \in [W]^2$, we set $\eta_i = \eta_W$. For each $x_i \in [V]^2 \setminus [W]^2$, we set $\eta_i = \eta_{\bar{W}}$. Let $\mu(W)$ and $\mu(\bar{W}) = \mu(V) - \mu(W)$ be the induced measures on the partitions, where $\mu(W) = \sum_{x_i \in [W]^2} \mu_0(x_i)$. Let η_W and $\eta_{\bar{W}}$ denote the partition representative values (the “centers”).

$$\eta_W = \sum_{x_i \in [W]^2} \frac{\mu_0(x_i)}{\mu(W)} c_0(x_i)$$

$$\eta_{\bar{W}} = \sum_{x_i \in \mathcal{X} \setminus [W]^2} \frac{\mu_0(x_i)}{\mu(\bar{W})} c_0(x_i)$$

After computing η_W and $\eta_{\bar{W}}$, we reassign x_i to W (set $\eta_i = \eta_W$), if $D_{KL}(\mu(x_i), \eta_W) < D_{KL}(\mu(x_i), \eta_{\bar{W}})$, otherwise $\eta_i = \eta_{\bar{W}}$. Then recompute η_W and $\eta_{\bar{W}}$. Repeat until convergence.

The Bregman clustering seeks to **minimize** the divergence between the two n -point distributions

$$\{\langle c_0(x_1), c_0(x_2), \dots, c_0(x_n) \rangle, \langle \eta_1, \eta_2, \dots, \eta_n \rangle\}$$

We, on the other hand, **maximize** the KL-divergence between the two 2-point distributions

$$\{\langle r(W), 1 - r(W) \rangle, \langle b(W), 1 - b(W) \rangle\}.$$

But the methods do not conflict with each other. Their η_W and $\eta_{\bar{W}}$ variables are akin to p and q in the derivation of the scan statistic. By minimizing their Bregman divergence, they are trying to allow η_W and $\eta_{\bar{W}}$ to be as close to variables they represent as possible (i.e. η_W should be close to each $c_0(x_i)$ for x_i defined in cluster W); and by maximizing our discrepancy we are separating p and q as much as possible, thus probabilistically representing the cluster edges and non-cluster edges more accurately with these ratios, respectively.

However, the Bregman divergence used by Banerjee et. al. [13, 14] typically assumes a less informative, uniform random graph model where $\mu_0(x_i) = \mu_0(x_j)$ for all i and j . Also when minimizing the KL-divergence, no edge at x_i would imply $c_0(x_i) = 0$, thus implying that the corresponding term of the KL-divergence, $c_0(x_i) \log \frac{c_0(x_i)}{\eta_i}$, is undefined. In their Bregman divergence model most similar to ours, this poses a problem as $c_0(x_i)$ can be 0 in our model; thus we do not compare the performance of these algorithms.

2.4 Algorithms

In this section, we describe two bottom-up, greedy clustering algorithms. For a graph $G = (V, E)$ there are $2^{|V|}$ possible clusters, That is, $|\mathcal{W}| \leq 2^{|V|}$. Clearly it is intractable to calculate discrepancy for every possible cluster through exhaustive search, as is often done with spatial scan statistics. We can, however, hope to find a locally optimal cluster. For an objective function $\Psi : 2^{|V|} \rightarrow \mathbb{R}$, define a *local maximum* as a subset $U \subseteq V$ such that adding or removing any vertex will decrease $\Psi(U)$. For some objective function Ψ

and two vertex sets U and W , define

$$\partial\Psi(U, W) = \begin{cases} \Psi(U \cup W) - \Psi(U) & W \subset V \setminus U \\ \Psi(U \setminus W) - \Psi(U) & W \subset U \end{cases}$$

where $\Psi(W) = d_P(W)$ for $W \subseteq \mathcal{W}$. Let U^+ (resp. U^-) be the set of vertices in $Lk(U)$ (resp. U) such that $\partial\Psi(U, v) > 0$ for each vertex v in $Lk(U)$ (resp. U). U_F^+ (resp. U_F^-) denotes the subset of U^+ (resp. U^-) that contains the fraction F of vertices with the largest $\partial\Psi(U, v)$ values. We now are set to describe two algorithms for refining a given subset U to find a local maximum in Ψ . Notice that both algorithms can be used to locally optimize any objective function, not limited to the Poisson discrepancy used here. They are similar to simple forms of simulated annealing where one fixes the temperature of the system while the other reduced temperature over time.

Greedy Nibble. The Greedy Nibble algorithm (Algorithm 1) alternates between an expansion phase and a contraction phase until the objective function cannot be improved. During expansion (resp. contraction) we iteratively add (resp. remove) the vertex that most improves the objective function until this phase can no longer improve the objective function.

Algorithm 1 Greedy-Nibble(U)

```

repeat
  expand = FALSE;   contract = FALSE
   $v^+ = \arg \max_{v \in Lk(U)} \partial\Psi(U, v)$ .
  while  $\partial\Psi(U, v^+) > 0$  do
    expand = TRUE
     $U = U \cup v^+$ .
     $v^+ = \arg \max_{v \in Lk(U)} \partial\Psi(U, v)$ .
  end while
   $v^- = \arg \max_{v \in U} \partial\Psi(U, v)$ .
  while  $\partial\Psi(U, v^-) > 0$  do
    contract = TRUE
     $U = U \setminus v^-$ .
     $v^- = \arg \max_{v \in U} \partial\Psi(U, v)$ .
  end while
until expand = FALSE and contract = FALSE

```

Greedy Chomp. The Greedy Chomp algorithm (Algorithm 2) is a more aggressive and faster version of the Greedy Nibble algorithm. Each phase adds a fraction F of the vertices which individually increase the Ψ value. If adding these $F|U^+|$ vertices simultaneously does not increase the overall Ψ value, then the fraction F is halved, unless $F|U^+| \leq 1$. Similar to simulated annealing, this algorithm makes very large changes to the subset at the beginning but becomes more gradual as it approaches a local optimum.

Algorithm 2 Greedy-Chomp(U)

```

repeat
  expand = FALSE;  $F = 1$ 
  Calculate  $U_F^+$ 
  while ( $\partial\Psi(U, U_F^+) < 0$  and  $F|U^+| \geq 1$ ) do
     $F = F/2$ ; Update  $U_F^+$ 
  end while
  while ( $\partial\Psi(U, U_F^+) > 0$ ) do
    expand = TRUE
     $U = U \cup U_F^+$ .
    Calculate  $U_F^+$ ;  $F = 1$ 
    while ( $\partial\Psi(U, U_F^+) < 0$  and  $F|U^+| \geq 1$ ) do
       $F = F/2$ ; Update  $U_F^+$ 
    end while
  end while
  contract = FALSE;  $F = 1$ 
  Calculate  $U_F^-$ 
  while ( $\partial\Psi(U, U_F^-) < 0$  and  $F|U^-| \geq 1$ ) do
     $F = F/2$ ; Update  $U_F^-$ 
  end while
  while ( $\partial\Psi(U, U_F^-) > 0$ ) do
    contract = TRUE
     $U = U \setminus U_F^-$ .
    Calculate  $U_F^-$ ;  $F = 1$ 
    while ( $\partial\Psi(U, U_F^-) < 0$  and  $F|U^-| \geq 1$ ) do
       $F = F/2$ ; Update  $U_F^-$ 
    end while
  end while
until (expand = FALSE and contract = FALSE)

```

Theorem 2.4.1. *Both the Greedy Nibble algorithm and the Greedy Chomp algorithm converge to a local maximum for Ψ .*

Proof. The algorithms increase the value of Ψ at each step, and there is a finite number of

subsets, so they must terminate. By definition the result of termination is a local maximum.

□

Variations. There are many possible heuristic variations on the above algorithms. More advanced simulated annealing can be used to direct the random walks in the graph, i.e. by selecting vertices with probability proportional to their contribution to the objective function.

In terms of initial seed selection, when time is not an issue, we recommend using each vertex as a seed. This ensures that every interesting cluster contains at least one seed. For larger graphs, randomly sampling some vertices as seeds should work comparably [98]. Clusters tend to be larger in this case, so most of them will still contain some seed. Alternatively, we could run another clustering algorithm to generate an initial seed and just use our greedy algorithms as a refinement.

In general, we use d_P as the objective function, but it is more prone to getting stuck in local maxima than is Q . Thus we enhance each initial seed by running the expansion phase of the algorithm with Q_2 since it closely approximates d_P as shown in Figure 2.4. We emphasize the importance of this variation in Appendix B.

Since our emphasis is on the discrepancy measurement rather than clustering technique, we focus on illustrating that these simple clustering techniques based on Poisson discrepancy find locally significant clusters.

Complexity It is difficult to analyze our algorithms precisely because they may alternate between the expansion and contraction phases many times. But Theorem 2.4.1 shows that this process is finite, and we notice that relatively few contraction steps are ever performed. Hence we focus on analyzing the worst case of the expansion phase in both algorithms.

Both algorithms are output dependent, where the runtime depends on the size of the final subset $|W|$ and the size of its neighborhood $|Lk(W)|$.

For Greedy Nibble we can maintain $Lk(W)$ and calculate v^+ in $O(|Lk(W)|)$ time. Thus the algorithm takes $O(|W| \cdot |Lk(W)|)$ time for each seed since v^+ needs to be calculated each iteration.

The Greedy Chomp algorithm could revert to the Greedy Nibble algorithm if F is immediately reduced to $1/|U^+|$ at every iteration. So worst case it is no faster than Greedy Nibble. In fact, each iteration takes $O(|Lk(W)| \log |Lk(W)|)$ time because the $\partial\Psi(U, v)$ values are sorted for all $v \in U^+$. However, in practice, a much smaller number of iterations are required because a large fraction of vertices are added at each iteration. If F were fixed throughout the algorithm, then we can loosely bound the runtime as $O(\log |W| \cdot |Lk(W)| \log |Lk(W)|)$. Since F is generally large when most of the vertices are added, this is a fair estimate of the asymptotics.

This analysis is further evaluated empirically in Section 2.5.

2.5 Analysis

This section focuses on empirically exploring four aspects of this work. First, we investigate the power and runtime of our algorithms. Second, we use Poisson discrepancy as a tool to evaluate and compare different clustering algorithms. Third, we investigate properties of the clusters found by our algorithms maximizing Poisson discrepancy. Fourth, we show that varying the γ parameter can give reliable estimates of the size of clusters and we examine cases when distinct relevant clusters overlap. Finally, throughout this analysis we demonstrate that maximizing Poisson discrepancy reveals interesting and relevant clusters in real-world graphs.

Runtime on Real-world Datasets We demonstrate the effectiveness of our algorithm on a variety of real world datasets, the sizes of which are summarized in Table 2.1.

The DBR dataset describes connections between threads (the set X) and users (the set Y) of the Duke Basketball Report message board from 2.18.07 to 2.21.07. Other datasets

include **Web**¹ which links websites and users, **Firm** which links AP articles and business firms, **Movie** which links reviewers and movies through positive reviews, and **Gene** which links genes and PubMed articles. In each case, high discrepancy clusters can be used to either provide advertising focus onto a social group or insight into the structure of the dataset.

Dataset	$ X $	$ Y $	$ E $	Nibble	Chomp
DBR	68	97	410	0.025	0.018
Web	1023	1008	4230	0.179	0.049
Firm	4950	7355	30168	9.377	0.251
Movie	1556	57153	1270473	-	32.91
Gene	6806	595036	1578537	-	242.7

TABLE 2.1: Sizes of real-world datasets and the average runtime in seconds for the Greedy Nibble and Greedy Chomp algorithms starting with singleton seeds. Runtimes for **Web** and **Firm** were generated with 100 random samples. Runtimes for **Movie** and **Gene** were generated with 50 random samples.

Power tests. Recall that the *power* of the test is the probability that the test statistic exceeds a critical value under some alternative hypothesis compared to some null hypothesis [57]. To calculate the power of our algorithm, we synthetically insert significant clusters into 100 random graphs and report the fraction of these graphs where our algorithm found the injected cluster.

In particular, we generate bipartite graphs using the Poisson random graph model such that $|X| = |Y| = 100$ and $|E| = 500$ where the expected degrees of vertices vary between 3 and 7. To inject a significant cluster, we choose a random set of vertices $W = X_W \cup Y_W$, where $X_W \subset X$, $Y_W \subset Y$, and $|X_W| = |Y_W| = 15$. We increase the probability that an edge is chosen between two vertices in X_W and Y_W by a factor of ρ . We scale the probabilities on the other pairs of vertices in the graph so that each vertex retains its original expected degree. By choosing an appropriate value of ρ , we can generate graphs with the same expected degree sequence and whose injected cluster is expected to be

¹ We thank Neilsen//Netratings, Inc., who provided the WEB dataset to us, for permission to use its data in this investigation.

significant. We repeat this process until we generate 100 graphs whose injected clusters have a p-value less than 0.05.

We run Greedy Nibble and Greedy Chomp using each vertex as a seed. We say that we successfully found the injected cluster W if the algorithm returns a cluster $\hat{W} = X_{\hat{W}} \cup Y_{\hat{W}}$ such that $sd(X_{\hat{W}}, X_W) \leq |X_W|$ and $d(Y_{\hat{W}}, Y_W) \leq |Y_W|$ and it either has p-value less than 0.05 or is among the top 5 clusters found.

We report the power of the algorithms in Table 2.2. It shows that 85% of the time Greedy Chomp locates the injected clusters. Note that we have used a relaxed criteria to determine when an injected cluster is found by our algorithm; a tighter qualification would reduce this power measurement.

Algorithm	Nibble	Chomp
Power	0.83	0.85

TABLE 2.2: Power for Greedy Nibble and Greedy Chomp tested on graphs of size 100×100 with an injected cluster of size 15×15 with p-value at most 0.05.

Algorithm Comparison. Poisson discrepancy provides an absolute measure of cluster significance. This allows comparison between different clustering of the same graph. We can evaluate the effectiveness of existing clustering algorithms by calculating the discrepancy of the clusters they find. Furthermore, we can enhance these clusters using Greedy Nibble or Greedy Chomp to maximize their discrepancy and evaluate how far from the local optimum these clusters used to be. We illustrate this by running the MCL algorithm [109] and the Ncut algorithm [106] on DBR. MCL generated 35 clusters and we fixed the number of clusters in Ncut to be 10. We report the top 4 clusters with the highest d_P value in Table 2.3. Ncut seems to find clusters with higher discrepancy. We then use clusters found by MCL and DBR as seed sets in the Greedy Chomp, further refining them in terms of their discrepancy. Ncut tends to do better than MCL in finding clusters within closer proximity of discrepancy local maxima.

MCL	0.0376	0.0248	0.0223	0.0211
MCL+Chomp	0.0667	0.0790	0.0620	0.0698
Ncut	0.0692	0.0529	0.0527	0.0473
Ncut+Chomp	0.0757	0.0688	0.0635	0.0713

TABLE 2.3: d_P values of top 4 clusters found with MCL and Ncut on DBR and the d_P values after their refinement with Greedy Chomp.

Cluster Overlap Analysis Many graph clustering methods partition the data into disjoint subsets, essentially making each a cluster. Our approach finds clusters which may overlap, and it considers the rest of the graph uninteresting instead of forcing it to be a cluster. We examine the top 6 clusters found from Greedy Nibble on DBR in an overlap matrix (Table 2.4). We use each vertex in X as a singleton seed set. The 1st, 2nd, 3rd and 5th clusters are very similar, representing a consistent set of about 13 threads on topics discussing the performance of players and strategy. The 4th cluster contains 14 threads which were posted by users who seem more interested in the site as a community and are more gossiping than analyzing. The 6th cluster contains an overlap of the above two types of topics and users: users who are interested in the community, but also take part in the analysis. The rest of the threads (about 60) deal with a wider and less focused array of topics.

C	1	2	3	4	5	6	d_P	p -value
1	13	12	12	1	12	8	0.0783	0.009
2	12	12	11	1	12	8	0.0764	0.019
3	12	11	14	0	11	7	0.0754	0.020
4	1	1	0	14	1	6	0.0749	0.020
5	12	12	11	1	13	8	0.0718	0.022
6	8	8	7	6	8	16	0.0703	0.077

TABLE 2.4: Overlap of threads among the top 6 clusters for DBR with their d_P and p -values found with the Greedy Nibble algorithm.

Discussion on $d_{P,\gamma}$ In the heuristic measure $d_{P,\gamma}$, γ serves as a resolution scale. For our algorithm, as γ varies, we observe an inverse linear correlation between γ and the average cluster size (Figure 2.5). We also show that as γ varies, our algorithm locates clusters that

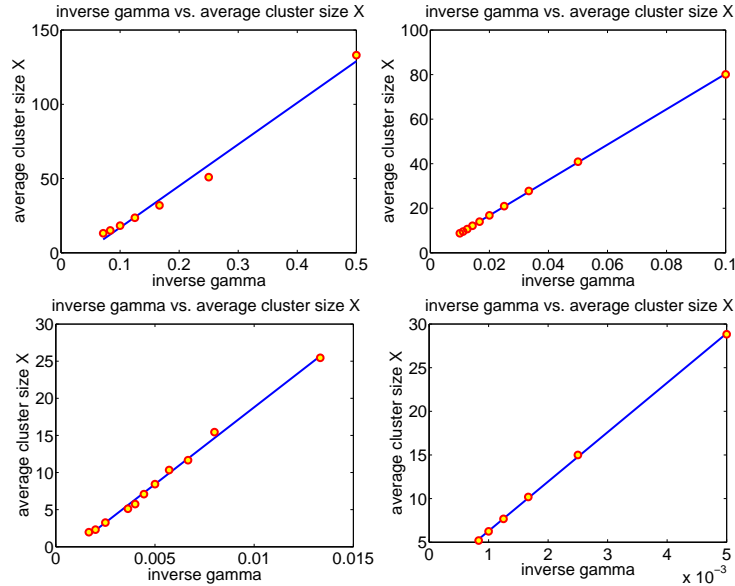


FIGURE 2.5: Plot of $1/\gamma$ vs. average cluster size on **Web** (top left), **Firm** (top right), **Movie** (bottom left), and **Gene** (bottom right).

are statistically significant on different scales, and that their contents remain meaningful.

This near-linear correlation makes γ a reliable resolution scale for our clustering algorithm. As γ goes to 0, the algorithm produces the whole graph as a cluster. As γ goes to infinity, the algorithm produces trivial singletons. The flexibility to modify γ allows a user to balance the importance of the statistical significance of the clusters found, maximized by $d_{P,\gamma}$, and their preferred size weighted by γ . This helps resolve issues (previously noted about modularity by [54]) about the preferred size of clusters which optimize d_P . For instance when searching for more focused clusters of smaller size, a reasonable γ weight can be easily inferred.

Manual evaluation of the results show that the contents of the clusters remain meaningful and useful as γ is varied. For example, the top clusters found on the **Movie** dataset with $\gamma = 200$ are shown to be popular box office movies in the 90's as they are consistently reviewed favorably by various reviewers. The top two clusters found on the **Gene** dataset with $\gamma = 200$ are genes in the UDP glucuronosyltransferase 1 and 2 family. The 4th ranked cluster consists of genes such as **MLH** and **PMS**, both are related to DNA mismatch repair.

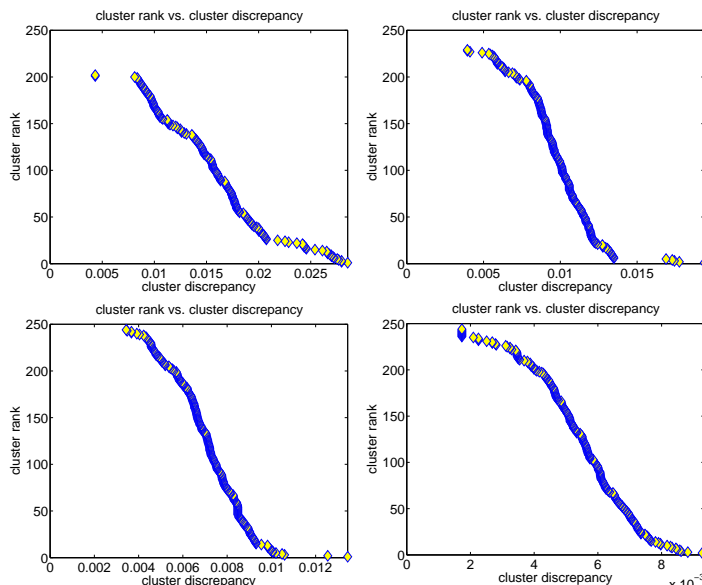


FIGURE 2.6: **Web**: cluster rank vs. cluster discrepancy with each X vertex used as a singleton seed set, $\gamma = 4$ (top left), $\gamma = 6$ (top right), $\gamma = 8$ (bottom left), $\gamma = 10$ (bottom right). Top ranked clusters appear at the bottom right of each figure.

The 8th ranked cluster for $\gamma = 200$ persists as the top ranked cluster when $\gamma = 600$; it consists of several genes for the zona pellucida glycoprotein, i.e ZP1 and ZP3A.

As γ increases, the nontrivial clusters with a high $d_{P,\gamma}$ -discrepancy should generally be much denser internally, since the ratio between the actual internal edges and expected edges should be greater than a given γ . On the other hand, clusters which persist as the top ranked clusters as γ increases are those that are most statistical significant in a dynamic setting. As γ increases, we would expect the number of such extremely anomalous clusters to decrease. For example, as shown in Figure 2.6 for the **Web** data set, as γ increases, the number of outlier clusters with comparatively very large discrepancy decreases. For $\gamma = 4$, many clusters seem to be significantly larger than the large component, while with $\gamma = 6$ and $\gamma = 8$ there are very few. Finally, with $\gamma = 10$ all clusters are basically in the same component.

The identification of clusters of varying size but consistently high statistical significance suggests that real-world networks are characterized by many different levels of granularity. This result is consistent with, e.g, the contrasting findings of [97] and [85],

where clusters of vastly different sizes but comparable modularities are detected in the same data set. This finding calls into question the wide variety of clustering methods which are only designed to detect one cluster for a given region or group of nodes, and a further study would be of interest.

2.6 Conclusions

The main contribution of this chapter is the introduction of a quantitative and meaningful measure, *Poisson discrepancy*, for clusters in graphs, derived from spatial scan statistics on point sets. According to our definition, the higher the discrepancy, the better the cluster. We identify interesting relations between Poisson discrepancy, local modularity, and Bregman divergences.

To illustrate the usefulness of this statistic, we describe and demonstrate two simple algorithms which find local optima with respect to the spatial scan statistic. In the context of real-world and synthetic datasets that are naturally represented as bipartite graphs, this method has identified individual clusters of vertices that are statistically the most significant. These clusters are the least likely to occur under a random graph model and thus best identify closely-related groups within the network. Our model places no restrictions on overlapping of clusters, thus allowing a data point to be classified into two or more groups to which it belongs. As our greedy algorithms are the simplest and most intuitive approach, it remains an open problem to find more effective algorithms to explore the space of potential subgraphs to maximize the Poisson discrepancy. Notice that Poisson discrepancy can also detect regions that are significantly under-populated by requiring $p < q$ in the alternative hypothesis.

Similarly the spatial scan statistic Bernoulli model for graph clustering can be derived from the corresponding model for point sets. However, this model requires that each potential edge be chosen with equal probabilities, regardless of the degree of a vertex.

Also, under this model each pair of vertices can have at most one edge.

For large graphs, what are the efficient ways to sample the different regions such that we get a good estimate of the significant clusters? Is there a similar construction as an overlap-kd tree in [83]? The spatial scan statistics is first introduced in a frequentist setting where we compute the p -values of a cluster through randomization; recent work by Neill et. al. [84] introduced the Bayesian spatial scan statistics by computing posterior probabilities of each potential cluster in close form and avoided randomization test. An interesting question is whether there are computational tractable Bayesian spatial scan statistics for graph clustering.

In summary, we argue that a graph cluster should be statistically justifiable, and a quantitative justification comes from a generalization of spatial scan statistics on graphs, such as the Poisson discrepancy.

Appendix to Chapter 2

A Proofs for the Properties of Graph Scan Statistic

The proof for Theorem 2.3.1 is as follows.

Proof. Since $\hat{W} \in V$ is the cluster that rejects H_0 under \mathcal{X} , we need to prove that the same cluster \hat{W} rejects H_0 under \mathcal{X}' .

Let $c(\hat{W})$ be the number of edges in $G(\hat{W})$. Let $c'(\hat{W})$ be the number of edges in $G'(\hat{W})$. Since $\forall x_i \in E(\hat{W}), x'_i = x_i$, edges are “shuffled around” (reassigned) in \mathcal{X}' comparing to \mathcal{X} by fixing edges in $E(\hat{W})$. We have $c(\hat{W}) \leq c'(\hat{W})$.

Let Λ and Λ' denote the test statistic for the two datasets \mathcal{X} and \mathcal{X}' . Since both \mathcal{X} and \mathcal{X}' have the same number of edges, the likelihood functions L_0 under H_0 are the same. The theorem is trivially true if $\Lambda = 1$.

If $\Lambda > 1$, that is, H_0 is rejected under \mathcal{X} , we just need to prove that $\Lambda' \geq \Lambda$ under \mathcal{X}' . Let $\mathcal{C} = \frac{c(V)^{-c(V)}}{\mu(V)}$, we have,

$$\begin{aligned}
 \Lambda &= \max_W \mathcal{C} \left(\frac{c(W)}{\mu(W)} \right)^{c(W)} \left(\frac{c(V) - c(W)}{\mu(V) - \mu(W)} \right)^{c(V) - c(W)} \\
 &= \mathcal{C} \left(\frac{c(\hat{W})}{\mu(\hat{W})} \right)^{c(\hat{W})} \left(\frac{c(V) - c(\hat{W})}{\mu(V) - \mu(\hat{W})} \right)^{c(V) - c(\hat{W})} \\
 &\leq \mathcal{C} \left(\frac{c'(\hat{W})}{\mu(\hat{W})} \right)^{c'(\hat{W})} \left(\frac{c(V) - c'(\hat{W})}{\mu(V) - \mu(\hat{W})} \right)^{c(V) - c'(\hat{W})} \\
 &\leq \max_W \mathcal{C} \left(\frac{c'(W)}{\mu(W)} \right)^{c'(W)} \left(\frac{c(V) - c'(W)}{\mu(V) - \mu(W)} \right)^{c(V) - c'(W)} \\
 &= \Lambda'
 \end{aligned}$$

The first inequality holds since for constants β, γ, τ , the function $g(x) = (\beta x)^x (\gamma(\tau - x))^{\tau - x}$ is an increasing function of x when $\beta x > \gamma(\tau - x)$. \square

We now present the proof for Theorem 2.3.2.

Proof. For an arbitrary W , let R_k denote the intersection of the critical region R and the subset of the sample space in which W is the most likely cluster. Let R'_k denote the intersection of the critical region R' and the subset of the sample space in which W is the most likely cluster. Let $A_k = \{(W, p, q) | p > q\}$.

We need to prove that Λ forms an individually most powerful test. We prove that if the first two statements in the definition is true, then the third statement does not hold. This is equivalent to prove the following:

- For any $\theta \in A_k$, $\beta'_k(\theta) \leq \beta_k(\theta)$.

To prove $\beta'_k(\theta) \leq \beta_k(\theta)$, for any $\theta \in A_k$, it is equivalent to prove the following:

$$\begin{aligned} & \beta'_k(\theta) - \beta_k(\theta) \\ &= P_\theta(X \in R'_k) - P_\theta(X \in R_k) \\ &= \Pr(X \in R'_k | \theta) - \Pr(X \in R_k | \theta) \\ &\leq 0. \end{aligned}$$

For an arbitrary W , define,

$$\begin{aligned} D_- &= \{x | x \in R_k, x \notin R'_k\} \\ D_+ &= \{x | x \in R'_k, x \notin R_k\} \end{aligned}$$

We define

$$M = \sup_{x \in D_+} \frac{L(\theta|x)}{L(\theta_0|x)},$$

where $\theta \in \Theta$ and $\theta_0 \in \Theta_0$.

By definition of D_+ and D_- , since R_k is defined by W , which is the most likely cluster in a subset of the sample space, we have that each x in D_- has a higher likelihood ratio than any x in D_+ , that is,

$$M = \sup_{x \in D_+} \frac{L(\theta|x)}{L(\theta_0|x)} \leq \inf_{x \in D_-} \frac{L(\theta|x)}{L(\theta_0|x)}$$

To prove the inequality, for any $\theta \in A_k$

$$\begin{aligned}
& \Pr(X \in R'_k|\theta) - \Pr(X \in R_k|\theta) \\
&= \Pr(X \in D_+|\theta) - \Pr(X \in D_-|\theta) \\
&= \int_{x \in D_+} f(x|\theta)dx - \int_{x \in D_-} f(x|\theta)dx \\
&= \int_{x \in D_+} L(\theta|x)dx - \int_{x \in D_-} L(\theta|x)dx \\
&= \int_{x \in D_+} \frac{L(\theta|x)}{L(\theta_0|x)} L(\theta_0|x)dx - \int_{x \in D_-} \frac{L(\theta|x)}{L(\theta_0|x)} L(\theta_0|x)dx \\
&\leq \int_{x \in D_+} ML(\theta_0|x)dx - \int_{x \in D_-} ML(\theta_0|x)dx \\
&= M \left(\int_{x \in D_+} L(\theta_0|x)dx - \int_{x \in D_-} L(\theta_0|x)dx \right) \\
&= M \left(\int_{x \in D_+} f(x|\theta_0)dx - \int_{x \in D_-} f(x|\theta_0)dx \right) \\
&= M (\Pr(X \in D_+|\theta_0) - \Pr(X \in D_-|\theta_0)) \\
&= M (\Pr(X \in R'_k|\theta_0) - \Pr(X \in R_k|\theta_0)) \\
&= M (\Pr(X \in R'|\theta_0) - \Pr(X \in R|\theta_0)) = 0
\end{aligned}$$

□

B Approximation of d_P

To demonstrate the propensity that our algorithm with d_P gets stuck in local minima, we ran our core algorithm with d_P , Q_1 , and Q_2 as objective functions. Then we report the d_P value of each cluster found, even when it was found using Q_1 or Q_2 . One would expect the clusters found with d_P to have the highest Poisson discrepancy, but this was not the case. Q_1 usually and Q_2 always outperformed d_P . In fact, the worst 20 clusters found using d_P had Poisson discrepancy values less than half the value of the worst clusters for

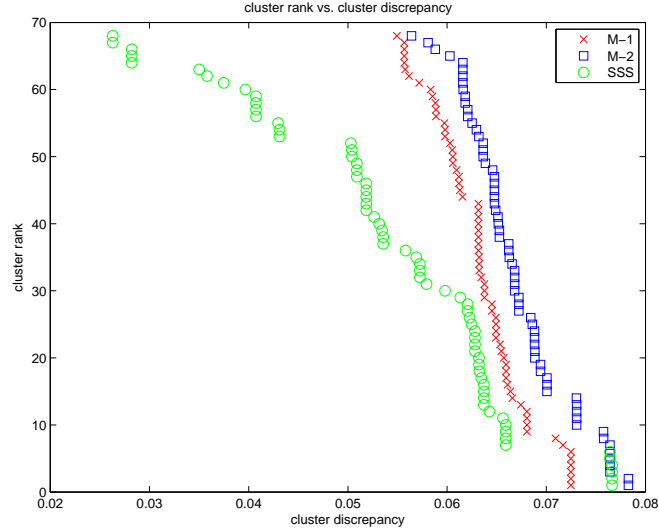


FIGURE 2.7: DBR: d_P values for all clusters running the algorithm seeded with each vertex from $|X|$ and using either Q_1 , Q_2 , and d_P as the objective function, represented as M-1 (red x), M-2 (blue square), and SSS (green circle), respectively.

Q_1 and Q_2 . We suspect that because d_P is “shallower” than Q_1 and Q_2 for low discrepancy subsets, the algorithm is more likely to get stuck in local minima.

p -value Test for DBR To calculate the p -value of a cluster S , we compare the cluster discrepancy $d_P(r_S, b_S)$ to the distribution of the highest discrepancy clusters from 1000 random graphs. The graphs are created under the Poisson fitted model. A cluster which has higher discrepancy than all but 50 random graphs has a p -value of $50/1000 = .05$. This indicates that only 5% of random graphs have a cluster as unexpected as the cluster found in our data set. Clusters with low p -values (usually less than .05) are said to be statistically significant, relative to d_P and our algorithm.

For the DBR data set, we rank clusters by their discrepancy and plot the p -values obtained by each, as shown in Figure 2.8, after removing duplicate clusters. The top 5 ranked clusters are statistically significant, with p -values between 0.009 and 0.022. We do not expect all the clusters found by our algorithm to display such high level of anomalousness. Since our core algorithm is based on seeds generated by Q_2 , we also plot the p -values of

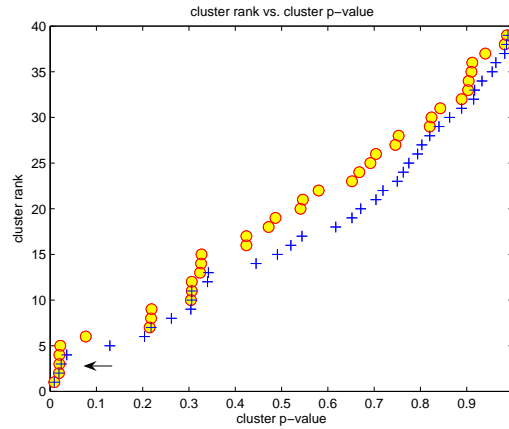


FIGURE 2.8: DBR: p -values for all clusters (yellow circle) and p -values for all seeds generated by Q_2 (blue plus). The top 5 ranked clusters with p -values between 0.009 and 0.022, are most statistically significant (follow arrow). The figure indicates that running our core algorithm with d_P further improves the discrepancy of our clusters.

the seeds, indicating the level of discrepancy improvement as a result of the core algorithm.

Chapter 3

Persistence in Expectation

This chapter is about the effect of “noise” on persistence diagrams as well as other statistical summaries relevant to persistence. The underlying motivation is to understand how the persistence-based statistics of a function with additive noise vary and to characterize expectations or averages of these variations. One goal of this is to denoise the function, subtract the persistence characteristics of the noise so that the persistence statistics of the function remain. Specifically, we study the statistical behavior of persistence diagrams, for constant functions with Gaussian noise defined on triangulations of topological spaces. The key quantities we study are pairing probabilities and the total persistence in expectation.

3.1 Introduction

Motivation. Given topological spaces and functions on them, persistent homology studies multi-scale features of the given data. It gives a notion of relative importance for features, in terms of the amount of change necessary to eliminate them [34]. Specifically, persistence diagrams capture this importance in a quantitative manner. In the case of noisy

data, possibly due to measurement inaccuracies or insufficient samplings, our goal is to quantify its uncertainty and further proceed with data denoising and smoothing. A first step towards this direction is to use total persistence computed from a persistence diagram as our summary statistics, and obtain a crude quantification of data uncertainty.

Prior work. Our work is driven by the objective to bring a statistical flavor to persistence, in aid of studying the topological properties of spaces with uncertainty. The stability of persistence diagrams lays a solid foundation for studying topological behaviors of functions under controlled perturbations [34, 36]. It provides a topological tool to understand how noise influences distances between persistence diagrams.

There has been related work that looks at persistent homology through the statistical lens. Niyogi, Smale and Weinberger combine homology and statistics and consider the case where a point cloud is drawn from a probability distribution that has support on or near a submanifold of \mathbb{R}^k [88]. They provide estimates on how many data points are needed to recover the homology of the submanifold from the point cloud with high confidence.

Bubenik and Kim compute the persistent homology of an unknown probability distribution that is assumed to belong to a parametric family of distributions [23]. Assuming a finite set of sampled points from the distribution, they demonstrate that using statistical estimators, the persistent homology of the unknown distribution can be recovered from the persistent homology of the simplicial complex constructed from the point sample. They also prove an upper bound on the expected distance between their persistence diagrams. Recent work by Bubenik et. al. gives an upper bound on the expected bottleneck distance between the persistence diagrams of the estimated function and that of the true function, under a nonparametric regression model [22].

Adler et. al. discuss manifold learning from random point cloud data and consider distributional properties of the barcodes of random field excursion sets [3]. Kahle studies the expectation and variance of the Betti numbers from Čech and Rips complexes built on

randomly sampled points in \mathbb{R}^k [67].

Contribution. The main contributions of this chapter to the theory of persistence are centered around four theorems.

- We derive combinatorially, the persistence pairing probabilities between vertices, for random piecewise-linear (PL) functions defined on a triangulation of \mathbb{S}^1 .
- We derive the expected total persistence, as a linear function in the size of the triangulation, for random PL functions defined on a triangulation of \mathbb{S}^1 .
- We give an upper bound on the expected total persistence, for random PL functions defined on a triangulation of a general topological space.
- We also give an upper bound on the expected change in total persistence, for PL functions with Gaussian perturbations, defined on a triangulation of a general topological space.

3.2 Preliminaries

In this section, we give a brief introduction to topological and statistical background needed to understand our main theorems. For details in persistent homology, see [50].

Triangulation and PL function. Let K be a *triangulation* of a triangulable topological space. It is defined as a finite *simplicial complex* together with a homeomorphism from the underlying space of the complex to the space. We define a function $f : V \rightarrow \mathbb{R}$ at all vertices V of K , where $|V| = n$. We assume f is *generic*, that is, all vertices have distinct function values. We obtain a piecewise-linear (PL) function $f : |K| \rightarrow \mathbb{R}$ using linear extension over the simplices. It is defined by $f(x) = \sum_i b_i(x)f(u_i)$, where the u_i are vertices of K and the $b_i(x)$ are barycentric coordinates of x [50].

Lower star filtration. Given a generic function $f : |K| \rightarrow \mathbb{R}$, we can order the vertices by increasing function values as $f(u_1) < f(u_2) < \dots < f(u_n)$. We then define K_i as the full sub-complex defined by the first i vertices, that is, a simplex σ belongs to K_i iff each vertex u_j of σ satisfies $j \leq i$. Recall the *star* $\text{St } u_i$ of a vertex u_i is the set of simplices that contain it, and the *lower star* $\text{St}_{-}u_i$ is the subset of simplicies for which u_i is the vertex with the maximum function value:

$$\text{St } u_i = \{\sigma \in K \mid u_i \in \sigma\};$$

$$\text{St}_{-}u_i = \{\sigma \in \text{St } u_i \mid x \in \sigma \Rightarrow f(x) \leq f(u_i)\}.$$

K_i is in fact the union of the first i lower stars. In other words, if $a_1 < a_2 < \dots < a_n$ are the function values of the vertices in K and $a_0 = -\infty$, then $K_i = K(a_i) = \bigcup_{u \in V, f(u) \leq a_i} \text{St}_{-}u$ for each i [50]. We therefore arrange an increasing sequence of complexes called the *lower star filtration* of f :

$$\emptyset = K_0 \subseteq K_1 \subseteq \dots \subseteq K_n = K.$$

It is shown in [50] that for $f(u_i) \leq a < f(u_{i+1})$, the sublevel set $|K|_a = f^{-1}(-\infty, a]$ is *homotopy equivalent* to K_i .

PL critical points. Recall the *link* $\text{Lk } v_i$ of a vertex v_i consists of all faces of simplices in the star that do not belong to the star, and the *lower link* $\text{Lk}_{-}v_i$ is the subset of simplicies in the link with smaller function values:

$$\text{Lk } v_i = \{\tau \subseteq \sigma \in \text{St } v_i \mid \tau \not\subseteq \text{St } v_i\};$$

$$\text{Lk}_{-}v_i = \{\sigma \in \text{Lk } v_i \mid x \in \sigma \Rightarrow f(x) < f(v_i)\}.$$

When we go from K_{i-1} to K_i , we attach the closed lower star of u_i along its lower link to the complex K_{i-1} . We define u_i as *PL regular* if its lower link is contractible, and *PL critical*, otherwise. We can classify the vertices using the *reduced Betti numbers* of their lower links [50]. Recall that $\tilde{\beta}_0$ is one less than β_0 , except for empty lower link, where

	$\tilde{\beta}_{-1}$	$\tilde{\beta}_0$	$\tilde{\beta}_1$
regular	0	0	0
minimum	1	0	0
saddle	0	1	0
maximum	0	0	1

TABLE 3.1: Classification of the vertices in a PL function on a 2-manifold.

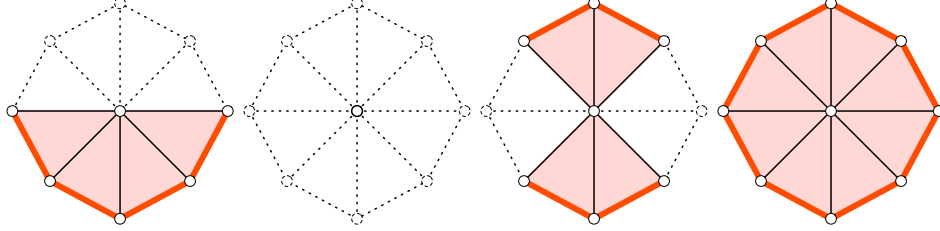


FIGURE 3.1: From left to right, in a 2-manifold: the lower star and lower link of a regular vertex, a minimum [50]. The shaded regions illustrate the underlying spaces of the lower star, while the thick solid lines illustrate the underlying spaces of the lower link.

$\tilde{\beta}_0 = \beta_0 = 0$ and $\beta_{-1} = 1$. This is shown in Table 3.1 [50]. Equivalently, a *minimum* is characterized by $\text{Lk}_-v_i = \emptyset$ and a *maximum* by $\text{Lk}_-v_i = \text{Lk } v_i$. We call u_i a *k-fold saddle* if its lower link consists of $k + 1 \geq 2$ paths, and we call u_i a *simple saddle* if $k = 1$. Some examples are shown in Figure 3.1.

Persistence. Given the lower star filtration, for each $i \leq j$, the inclusion map $K_i \rightarrow K_j$ induces homomorphisms between **homology groups**, $f_p^{i,j} : H_p(K_i) \rightarrow H_p(K_j)$, for dimension p . The sequence of homology groups connected by homomorphisms is therefore,

$$0 = H_p(K_0) \rightarrow H_p(K_1) \rightarrow \dots \rightarrow H_p(K_n) = H_p(K).$$

The p -th **persistent homology groups** are defined as $H_p^{i,j} = \text{im } f_p^{i,j}$ for $0 \leq i, j \leq n$. We have a **birth** at K_i if the map $f_p^{i-1,i}$ is not surjective and a **death** at K_j is the map $f_p^{j-1,j}$ is not injective. Furthermore, if γ is born at K_i , then it dies entering K_j if it merges with an older class as we go from K_{j-1} to K_j , that is, $f_p^{i,j-1}(\gamma) \notin H_p^{i-1,j-1}$ and $f_p^{i,j}(\gamma) \in H_p^{i-1,j}$ [50]. If γ is born at K_i and dies entering K_j , its **persistence** is defined as $a_j - a_i$.

The corresponding simplex σ_i entering K_i is then paired with the simplex σ_j entering K_j , representing the birth and death of the homology class γ . By running the ordinary

persistence algorithm [51], we obtain the pairings of simplices in K . If two simplices in the same lower-star are paired they therefore represent the homology class that dies immediately after birth. Only pairings between simplices in different lower stars carry significance. Some simplices remain unpaired as they create homology classes that never dies, so-called *essential simplices*. They are paired by the extended persistence algorithm [35]. In subsequent sections, we use ordinary persistence only when discussing pairing probabilities. Otherwise, we use extended persistence.

It is convenient to use a piecewise constant approximation $\bar{f} : K \rightarrow \mathbb{R}$ of $f : |K| \rightarrow \mathbb{R}$ with $\bar{f}(\sigma) = \max_{x \in \sigma} f(x)$. Suppose f is generic, then ordering simplices by their values under \bar{f} and breaking the ties by dimension gives the lower star filtration of f [79]. This implies that \bar{f} and f give the same persistence diagrams. Therefore we replace the two simplices (σ_i, σ_j) that are paired, with their unique highest vertices and speak of pairings between vertices (v_i, v_j) , where $\bar{f}(\sigma_i) = v_i$, $\bar{f}(\sigma_j) = v_j$. Since \bar{f} and f agree on vertices and give the same persistence diagrams, unless otherwise specified, when we say vertices are paired by running persistence algorithm on f , it is understood that we use a piecewise constant approximation.

We construct the p -th *persistence diagram* $\text{Dgm}_p(f)$ as a multiset of points in the *extended plane*, where each point represents the birth and death of a p -dimensional homology class. For each pair of vertices (v_i, v_j) which represent the birth and death of a p -dimensional homology class, the diagram contains the point $x = (f(v_i), f(v_j))$, whose persistence is $\text{pers}(x) = f(v_j) - f(v_i)$. Points in the diagrams can have non-negative integer multiplicities.

We define the *total persistence* of f as the sum of the persistence of all points in $\text{Dgm}_p(f)$, for all dimensions. That is,

$$\text{Pers}(f) = \sum_p \sum_{x \in \text{Dgm}_p(f)} \text{pers}(x),$$

where a point contributes as many times as it occurs.

We define the *multiplicity* k of a vertex $u \in K$ as the number of critical vertices that are paired with u during the persistence pairing. If u is regular, $k = 0$. If u is a local minimum, a local maximum or a simple saddle, $k = 1$. If u is a k -fold saddle, its multiplicity is k . u can be paired with k_1 local minima (and/or saddles) and k_2 local maxima (and/or saddles) at the same time, where $k_1, k_2 \geq 0$ and $k_1 + k_2 \leq k$.

Counting. To prove our main theorem, we use a simple formula in counting. Given an ordered list with a items, x_1, x_2, \dots, x_a , we insert b unordered items, y_1, y_2, \dots, y_b . Then the number of possible outcomes of this process is $\frac{(a+b)!}{a!}$.

To obtain the formula, we can insert y_1 before x_1 or after x_a , or between x_i and x_{i+1} , for $1 \leq i \leq a - 1$. There are $a + 1$ possible locations. To insert y_2 after inserting y_1 , there are $a + 2$ possible locations. To insert y_b , there are $a + b$ possible locations. In summary, we have $(a + 1)(a + 2)\dots(a + b) = \frac{(a+b)!}{a!}$ possible orderings.

Normal distribution. Several properties of the **normal** distribution are key ingredient in proving our theorems [62, 91, 64].

Let X be a random variable drawn from $N(0, \sigma^2)$, that is, $X \sim N(0, \sigma^2)$. Let $Y = |X|$. Then Y is **half-normal** distributed, and $E[Y] = \sqrt{\frac{2}{\pi}}\sigma$.

Let $\bar{X} = \{X_1, X_2, \dots, X_d, X_{d+1}\}$ ($d \geq 2$) be random variables drawn independently and identically (i.i.d.) from $N(0, 1)$, that is, $\bar{X} \stackrel{i.i.d.}{\sim} N(0, 1)$. We obtain the order statistics by sorting,

$$X_{1:d+1} \geq X_{2:d+1} \geq \dots \geq X_{d:d+1} \geq X_{d+1:d+1},$$

where $X_{i:d+1}$ is the i -th largest value in \bar{X} . Define $S_{i:d+1} = E[X_{i:d+1}] - E[X_{i+1:d+1}]$ as the expected *spacing*, $1 \leq i \leq d$. We obtain a set of spacings $S = \{S_{1:d+1}, S_{2:d+1}, \dots, S_{d:d+1}\}$.

We have the following properties associated with the order statistics:

- (Expected order statistics are symmetric) $E[X_{i:d+1}] = -E[X_{d-i+2:d+1}]$;
- (Spacings are symmetric) $S_{i:d+1} = S_{d-i+1:d+1}$.
- $S_{1:d+1} > S_{2:d+1} > \dots > S_{\lfloor \frac{d+1}{2} \rfloor : d+1}$.
- $S_{1:d+1} > S_{1:d+2}$.

Specifically, let X_1, X_2 and X_3 be random variables drawn i.i.d. from $N(0, 1)$. We obtain the order statistics, $X_{1:3} \geq X_{2:3} \geq X_{3:3}$. Define $C_0 = \frac{1}{3}(S_{1:3} + S_{2:3}) = \frac{1}{3}(E[X_{1:3}] - E[X_{3:3}])$. It has been shown that, $E[X_{1:3}] \approx 0.84628$, $E[X_{2:3}] \approx 0.00000$, $E[X_{3:3}] \approx -0.84628$ [62]. Therefore, $C_0 \approx 0.56419$.

3.3 Pairing Probabilities

We are interested in the pairing structure by running the ordinary persistence algorithm on a PL function. Our first theorem, Theorem 3.3.1, gives a combinatorial formula for the persistence pairing probabilities between vertices, for random PL functions defined on a triangulation of \mathbb{S}^1 . Specifically, we look at the class of functions f whose values are drawn i.i.d. from some distributions.

Theorem 3.3.1 (Pairing Probabilities). *Let K be a triangulation of \mathbb{S}^1 with vertex set $V = \{u_1, u_2, \dots, u_n\}$, ordered counter-clockwise, while $u_i = u_{i+n}$. Let $f : V \rightarrow \mathbb{R}$ be a function defined on V such that $f(u_i)$ is drawn i.i.d. from some distribution. Assume f is generic. We obtain a PL function $f : |K| \rightarrow \mathbb{R}$ using linear extension over the simplices. Let $Q(i, j)$ denote the probability that u_i and u_j are paired by running the ordinary persistence algorithm on f , where u_i and u_j correspond to the birth and death of a homology class respectively. Then for any integer $c > 0$,*

$$Q(i, i+c) = \sum_{j=1}^{n-c-2} \frac{j(j+c-2)!}{(j+c+2)!} + \sum_{k=1}^{c-2} \frac{k(k+n-c-2)!}{(k+n-c+2)!}.$$

Meanwhile, $Q(i, j) = Q(i, n+2-j)$, $Q(i, j) = Q(j, i)$.

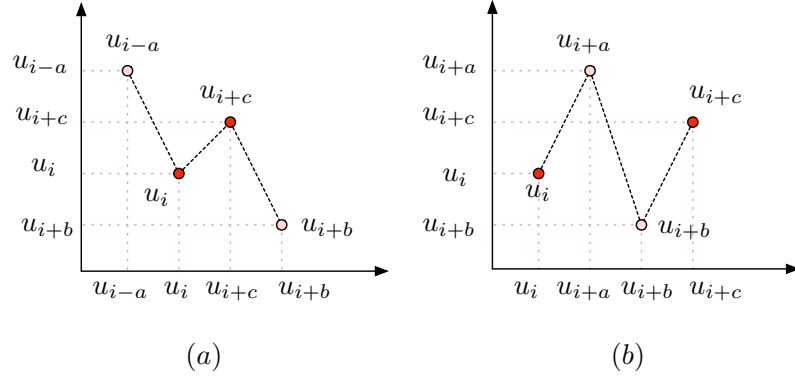


FIGURE 3.2: Two cases in the proof of Theorem 3.3.1.

Proof. Let $Q(i, i + c)$ denote the probability that u_i and u_{i+c} are paired, where u_i and u_{i+c} correspond to the birth and death of a homology class. This implies $f(u_i) < f(u_{i+c})$.

We refer to the counter-clockwise ordering of the u_i as the *index ordering*. We refer to the ordering of the u_i by increasing function values as the *height ordering*.

Suppose all points are drawn from some distribution and that all values are distinct. All $n!$ possible height orderings occur with equal probability.

To compute $Q(i, i + c)$, we count the index orderings (or height orderings) in which u_i and u_{i+c} are paired. The key to our proof is as follows. Suppose u_i and u_{i+c} are paired. We consider the process of inserting $c - 1$ vertices into the gap between u_i and u_{i+c} , and the remaining vertices between u_{i+c} and u_i , counterclockwise in the index ordering. We add constraints to the heights of the inserted vertices, so that their insertions do not affect the (u_i, u_{i+c}) pair. $Q(i, i + c)$ is the total number of possible orderings divided by $n!$.

There are two cases, as shown in Figure 3.2.

- (a) Vertices inserted between u_i and u_{i+c} form pairs among themselves, with persistence smaller than $|f(u_i) - f(u_{i+c})|$.
- (b) Vertices inserted between u_{i+c} and u_i form pairs among themselves, with persistence smaller than $|f(u_i) - f(u_{i+c})|$.

Case (a): Suppose u_i and u_{i+c} remain paired as we insert vertices between them. There

exist vertices u_{i-a} and u_{i+b} that gives an index ordering,

$$u_{i-a}, \dots, u_i, \dots, u_{i+c}, \dots, u_{i+b}, \quad (3.1)$$

and a height ordering,

$$u_{i+b}, \dots, u_i, \dots, u_{i+c}, \dots, u_{i-a}, \quad (3.2)$$

where u_{i-a} is the first vertex to the left of u_i s.t. $f(u_{i-a}) > f(u_{i+c})$, and u_{i+b} is the first vertex to the right of u_{i+c} s.t. $f(u_{i+b}) < f(u_i)$. For u_{i-a} and u_{i+b} to exist, we have $a \geq 1$, $b \geq c + 1$ and $a + b + 1 \leq n$.

We look at all possible insertions into the index ordering (3.1) s.t. u_i and u_{i+c} remain paired. We proceed as follows:

- (i) insert $a - 1$ vertices between u_{i-a} and u_i s.t. for each inserted vertex x , $f(u_i) < f(x) < f(u_{i+c})$;
- (ii) insert $c - 1$ vertices between u_i and u_{i+c} s.t. for each inserted vertex x , $f(u_i) < f(x) < f(u_{i+c})$;
- (iii) insert $b - c - 1$ vertices between u_{i+c} and u_{i+b} s.t. for each inserted vertex x , $f(u_i) < f(x) < f(u_{i+c})$;
- (iv) insert $n - a - b - 1$ vertices between u_{i+b} and u_{i-a} .

Equivalently the above procedure can be viewed in terms of insertions into the height ordering (3.2):

- (1) insert $a + b - 3$ vertices between u_i and u_{i+c} which gives a height ordering

$$u_{i+b}, u_i, x_1, x_2, \dots, x_{a+b-3}, u_{i+c}, u_{i-a}, \quad (3.3)$$

this is done by combining cases (i), (ii) and (iii);

- (2) insert $n - a - b - 1$ vertices into the height ordering (3.3).

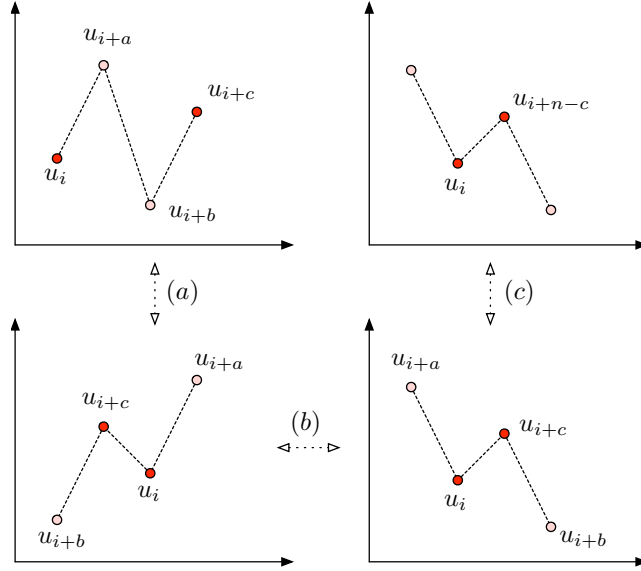


FIGURE 3.3: (a) Rotation, (b) reflection, (c) re-indexing by replacing c with $n - c$.

For fixed a and b , step (1) gives $(a + b - 3)!$ orderings, and step (2) gives $\frac{n!}{(a+b+1)!}$ orderings. Step (1) and (2) combined gives $(a + b - 3)! \frac{n!}{(a+b+1)!}$ orderings.

The total number of orderings counted in case (a) is

$$\sum_{a,b} (a + b - 3)! \frac{n!}{(a + b + 1)!},$$

where $a \geq 1$, $b \geq c + 1$ and $a + b + 1 \leq n$. This can be further simplified to

$$n! \sum_{j=1}^{n-c-2} \frac{j(j + c - 2)!}{(j + c + 2)!}. \quad (3.4)$$

Case (b): by symmetry, as shown in Figure 3.3, replacing c in equation 3.4 with $n - c$ gives the total number of orderings,

$$n! \sum_{k=1}^{c-2} \frac{k(k + n - c - 2)!}{(k + n - c + 2)!}. \quad (3.5)$$

Therefore, equation 3.4 and equation 3.5 combined gives the number of height orderings where u_i and u_{i+c} remain paired. It is then divided by $n!$ to give $Q(i, i + c)$. \square

Some examples of pairing probabilities when $n = 3, 4, 5, 6, 7$ are shown in Table 3.3, where $Q(1, :)$ includes all pairing probabilities between vertex u_1 and all other vertices, that is, $Q(1, 1), Q(1, 2), \dots, Q(1, n)$.

n	$Q(1, :)$
3	$[0, 0, 0]/3!$
4	$[0, 1, 0, 1]/4!$
5	$[0, 7, 1, 1, 7]/5!$
6	$[0, 48, 104, 10, 48]/6!$
7	$[0, 360, 88, 32, 32, 88, 360]/7!$

TABLE 3.2: Examples of Q .

3.4 Expected Total Persistence

The remaining three theorems involve various forms of the total persistence in expectation. Theorem 3.4.1 gives a close-form formula for the expected total persistence for random PL functions defined on a triangulation of \mathbb{S}^1 . Theorem 3.4.3 generalizes the result to a triangulation of a general topological space, by giving an upper bound for the expected total persistence. Theorem 3.5.1 (proved in a subsequent section) gives the expected difference, between the total persistence of a PL function, and that of the PL function with Gaussian perturbations.

Theorem 3.4.1 (Expected Total Persistence for \mathbb{S}^1). *Let K be a triangulation of \mathbb{S}^1 with n vertices, $V = \{u_1, u_2, \dots, u_n\}$. Let $f : V \rightarrow \mathbb{R}$ be a function defined on V such that, $\{f(u_1), f(u_2), \dots, f(u_n)\} \stackrel{i.i.d.}{\sim} N(0, 1)$. We then obtain by linear extension a PL function $f : |K| \rightarrow \mathbb{R}$. Vertices in K are paired by running the extended persistence algorithm on f . Then,*

$$E[\text{Pers}(f)] = C_0 \cdot n,$$

where $C_0 = \frac{1}{3}(S_{1:3} + S_{2:3}) \approx 0.56419$.

To prove Theorem 3.4.1, we start by proving the following lemma that takes a close look at the persistence pairing structure. The lemma holds for general simplicial complexes and we will apply the result for triangulations of \mathbb{S}^1 and general topological spaces.

Lemma 3.4.2. *Let K be a simplicial complex where d_{max} is the maximum degree of its vertices. Let $f, g : V \rightarrow \mathbb{R}$ be functions defined on V such that $f = g$ except for a vertex $u \in K$ such that $f(u) = a$ and $g(u) = b$, while $b > a$. We then obtain by linear extensions PL functions $f, g : |K| \rightarrow \mathbb{R}$. Then,*

$$\text{Pers}(g) - \text{Pers}(f) \leq d_{max}(b - a).$$

Proof. As we continuously change the function value at u from a to b , this corresponds to a number of transpositions of consecutive vertices involving u in the ordering defining the lower-star filtration. During a transposition of two consecutive vertices, the pairs can switch vertices only at moments when these vertices have the same value [38].

Let $f_0(u) = a$, $f_1(u) = b$. Suppose that we continuously change the function value at u through a straight-line homotopy,

$$f_\lambda(u) = (1 - \lambda)f_0(u) + \lambda f_1(u).$$

There are m values of λ for which f_λ is not injective. These are the values when transpositions happen between vertex u and vertex v , where $f_\lambda(u) = f(v)$. Adding $\lambda_0 = 0$ and $\lambda_{m+1} = 1$, we get the following ordered list of these values: $\lambda_0 < \lambda_1 < \dots < \lambda_m < \lambda_{m+1}$.

First, we consider two values without transposition between them, $\lambda_i < r < s < \lambda_{i+1}$. The persistence pairing is the same for f_r and for f_s . Let k be the multiplicity of u . k is the same for f_r and f_s as well. We have four cases,

- Case 1. If u is a local minimum ($k = 1$), then (u, v) is a member of the pairing for both f_r and f_s . Since $f_r(v) = f_s(v)$, the change in total persistence from r to s is $f_r(u) - f_s(u)$, which is negative.

- Case 2. If u is a local maximum ($k = 1$), then (w, u) is a member of the pairing for both f_r and f_s . Since $f_r(w) = f_s(w)$, the change in total persistence from r to s is $f_s(u) - f_r(u)$, which is positive.
- Case 3. If u is regular ($k = 0$), then the change in total persistence is 0.
- Case 4. If u is a k -fold saddle ($k \geq 1$), u can be paired with k_1 local minima (and/or saddles) and k_2 local maxima (add/or saddles) at the same time, where $k_1, k_2 \geq 0$ and $k_1 + k_2 \leq k$. Then $(w_1, u), (w_2, u), \dots, (w_{k_1}, u)$ and $(u, v_1), (u, v_2), \dots, (u, v_{k_2})$ are members of the pairing for both f_r and f_s . The change in total persistence from r to s is

$$\begin{aligned}
k_1(f_s(u) - f_r(u)) + k_2(f_r(u) - f_s(u)) &= (k_1 - k_2)(f_s(u) - f_r(u)) \\
&\leq k(f_s(u) - f_r(u)) \\
&\leq d_{max}(f_s(u) - f_r(u)).
\end{aligned}$$

In summary, all four cases above give a change in total persistence less than or equal to $d_{max}(f_s(u) - f_r(u))$.

Second, a transposition changes the matching of vertices but does not change the total persistence.

As we change the function value at u through the straight-line homotopy,

$$\begin{aligned}
\text{Pers}(g) - \text{Pers}(f) &\leq \sum_{j=0}^m d_{max}(f_{\lambda_{j+1}}(u) - f_{\lambda_j}(u)) \\
&= d_{max}(f_1(u) - f_0(u)) \\
&= d_{max}(b - a).
\end{aligned}$$

□

We now prove Theorem 3.4.1, based on cases discussed in the previous lemma.

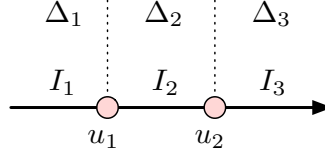


FIGURE 3.4: Sort vertices in the lower link of u by increasing function value.

Proof. For a random vertex $u \in K$, define its contribution to the total persistence as Δ . We would like to prove that $\mathbb{E}[\Delta] = C_0$.

Since K is a triangulation of \mathbb{S}^1 , u has two vertices in its link, u_1 and u_2 . We order them by increasing function value. Without loss of generality, we assume $f(u_1) < f(u_2)$. $f(u_1)$ and $f(u_2)$ partition \mathbb{R} into 3 intervals, $I_1 = (-\infty, f(u_1))$, $I_2 = [f(u_1), f(u_2))$ and $I_3 = [f(u_2), +\infty)$, as shown in Figure 3.4.

The probability that $f(u)$ falls in any interval is, $\Pr[f(u) \in I_i] = \frac{1}{3}$, for $1 \leq i \leq 3$. When $f(u) \in I_i$, we define Δ_i as its contribution to the total persistence. We have $\mathbb{E}[\Delta] = \frac{1}{3} \sum_{i=1}^3 \mathbb{E}[\Delta_i]$.

Suppose that u is inserted into the triangulation, with function value $f(u)$, where $f(u) \in I_i$. We would like to compute its contribution to the total persistence due to the insertion. We carefully define functions at u , $f_0(u)$ and $f_1(u)$, such that u at $f_0(u)$ has 0 contribution to the total persistence, and $f_1(u) = f(u)$. We then define a straight-line homotopy at u by continuously changing its function value from $f_0(u)$ to $f_1(u)$, that is,

$$f_\lambda(u) = (1 - \lambda)f_0(u) + \lambda f_1(u).$$

This corresponds to a number of transpositions of consecutive vertices involving u in the ordering defining the lower-star filtration. As a result, the pairing and the total persistence might change as well. The homotopy from f_0 to f_1 offers a proper framework to track the accumulation of contributions to the total persistence. We have the same four cases discussed in Lemma 3.4.2.

We compute $\mathbb{E}[\Delta_i]$ as follows,

(1) If $f(u) \in I_2$, u is regular, $\Delta_2 = 0$.

(2) If $f(u) \in I_1$, u is a local minimum. Let $f_0(u) = f(u_1)$, $f_1(u) = f(u)$.

$$\Delta_1 = \sum_{j=0}^m f_{\lambda_j}(u) - f_{\lambda_{j+1}}(u) = f(u_1) - f(u),$$

$$\mathbb{E}[\Delta_1] = \mathbb{E}[f(u_1) - f(u)] = \mathbb{E}[X_{2:3}] - \mathbb{E}[X_{3:3}] = S_{2:3}.$$

(3) If $f(u) \in I_3$, u is a local maximum. Let $f_0(u) = f(u_2)$, $f_1(u) = f(u)$.

$$\Delta_3 = \sum_{i=0}^m f_{\lambda_{j+1}}(u) - f_{\lambda_j}(u) = f(u) - f(u_2),$$

$$\mathbb{E}[\Delta_3] = \mathbb{E}[f(u) - f(u_2)] = \mathbb{E}[X_{1:3}] - \mathbb{E}[X_{2:3}] = S_{1:3}.$$

Therefore,

$$\mathbb{E}[\Delta] = \frac{1}{3} \sum_{i=1}^3 \mathbb{E}[\Delta_i] = \frac{1}{3}(S_{1:3} + S_{2:3}) = C_0$$

□

Next, we extend the above result to a triangulation of a general topological space, and prove the more complicated upper bound on total persistence in Theorem 3.4.3.

Theorem 3.4.3 (Expected Total Persistence for a General Topological Space). *Let K be a triangulation of a general topological space with n vertices. Let d_{min} and d_{max} be the minimum and maximum degree of its vertices. Vertices in K are paired by running the extended persistence algorithm on f . Then,*

$$\mathbb{E}[\text{Pers}(f)] \leq C_1 \cdot n,$$

where $C_1 = \frac{1}{3} \cdot S_{1:d_{min}+1} \cdot \lfloor \frac{d_{max}}{2} \rfloor (\lfloor \frac{d_{max}}{2} \rfloor + 1)$.

Proof. For a random vertex $u \in K$, define Δ as its contribution to the total persistence.

We would like to prove that $\mathbb{E}[\Delta] \leq C_1$.

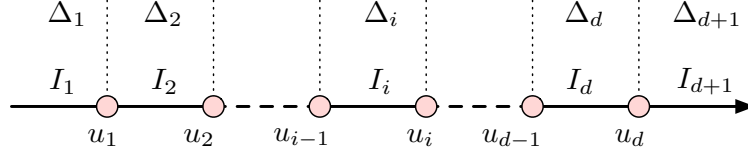


FIGURE 3.5: Sort vertices in the lower link of u by increasing function value.

Let the degree of u be d . We order vertices in its link by increasing function value, w.l.o.g., assume $f(u_1) < f(u_2) < \dots < f(u_d)$. As shown in Figure 3.5, the $f(u_i)$ partitions \mathbb{R} into $d + 1$ intervals, where $I_1 = (-\infty, f(u_1))$, $I_{d+1} = [f(u_d), +\infty)$, $I_i = [f(u_{i-1}), f(u_i))$ for $2 \leq i \leq d$. The probability that $f(u)$ falls in any interval is

$$\Pr[f(u) \in I_i] = \frac{1}{d+1}.$$

When $f(u) \in I_i$, we define Δ_i as its contribution to the total persistence. We need to compute

$$\mathbb{E}[\Delta] = \frac{1}{d+1} \sum_{i=1}^{d+1} \mathbb{E}[\Delta_i].$$

Suppose that we continuously change the function value at u through a straight-line homotopy from $f_0(u)$ to $f_1(u)$. Again, we have the same four cases discussed in Lemma 3.4.2.

We compute $\mathbb{E}[\Delta_i]$ as follows.

- (1) If $f(u) \in I_2$ or $f(u) \in I_d$, u is regular, $\Delta_2 = \Delta_d = 0$.
- (2) If $f(u) \in I_1$, u is a local minimum. Let $f_0(u) = f(u_1)$, $f_1(u) = f(u)$.

$$\Delta_1 = \sum_{j=0}^m f_{\lambda_j}(u) - f_{\lambda_{j+1}}(u) = f(u_1) - f(u),$$

$$\mathbb{E}[\Delta_1] = \mathbb{E}[f(u_1) - f(u)] = \mathbb{E}[X_{d:d+1}] - \mathbb{E}[X_{d+1:d+1}] = S_{d:d+1} = S_{1:d+1}.$$

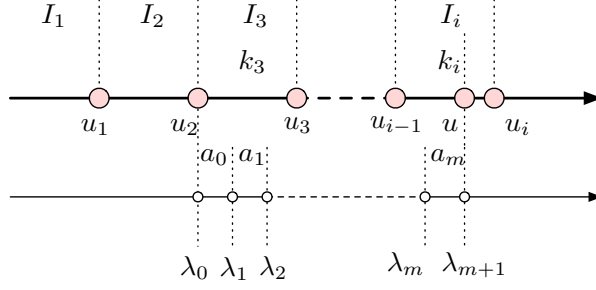


FIGURE 3.6: Relative multiplicities between intervals.

- (3) If $f(u) \in I_{d+1}$, u is a local maximum. Let $f_0(u) = f(u_d)$, $f_1(u) = f(u)$.

$$\Delta_{d+1} = \sum_{i=0}^m f_{\lambda_{j+1}}(u) - f_{\lambda_j}(u) = f(u) - f(u_d),$$

$$E[\Delta_{d+1}] = E[f(u) - f(u_{d+1})] = E[X_{1:d+1}] - E[X_{2:d+1}] = S_{1:d+1}.$$

- (4) If $f(u) \in I_i$, where $3 \leq i \leq \lfloor \frac{d}{2} \rfloor + 1$, let $f_0(u) = f(u_2)$, $f_1(u) = f(u)$. Define the multiplicity of u as k_l when $f_{\lambda_l}(u) \in I_l$, for $3 \leq l \leq i$. We have, as illustrated in Figure 3.6,

$$\Delta_i \leq k_3(f(u_3) - f(u_2)) + k_4(f(u_4) - f(u_3)) + \dots + k_i(f(u) - f(u_{i-1})).$$

Since $k_3, k_4, \dots, k_i \leq i - 2$,

$$\Delta_i \leq (i - 2)(f(u) - f(u_2)).$$

Then,

$$\begin{aligned} E[\Delta_i] &\leq (i - 2)(E[f(u)] - E[f(u_2)]) \\ &= (i - 2)(E[X_{d-i+2:d+1}] - E[X_{d:d+1}]) \\ &= (i - 2)(E[X_{2:d+1}] - E[X_{i:d+1}]) \\ &= (i - 2)(S_{2:d+1} + S_{3:d+1} + \dots + S_{i-1:d+1}). \end{aligned}$$

Since $S_{1:d+1} > S_{2:d+1} > \dots > S_{i-1:d+1}$,

$$\begin{aligned} \mathbb{E}[\Delta_i] &\leq (i-2)((i-2)S_{2:d+1}) \\ &= (i-2)^2 S_{2:d+1} \\ &\leq (i-2)^2 S_{1:d+1}. \end{aligned}$$

(5) By symmetry, If $f(u) \in I_i$, where $\lfloor \frac{d}{2} \rfloor + 1 \leq i \leq d-1$, $\mathbb{E}[\Delta_i] = \mathbb{E}[\Delta_{d+2-i}]$.

Summary. Let $p = \lfloor \frac{d}{2} \rfloor$, We have,

$$\begin{aligned} \mathbb{E}[\Delta] &= \frac{1}{d+1} \sum_{i=1}^{d+1} \mathbb{E}[\Delta_i] \\ &\leq \frac{1}{d+1} (2S_{1:d+1} + 2S_{1:d+1} \sum_{i=3}^{p+1} (i-2)^2) \\ &= \frac{2S_{1:d+1}}{d+1} (1 + \sum_{i=1}^{p-1} i^2) \\ &\leq \frac{2S_{1:d+1}}{d+1} \sum_{i=1}^p i^2 \\ &= \frac{2S_{1:d+1}}{d+1} \cdot \frac{p(p+1)(2p+1)}{6} \\ &= \frac{1}{3} \cdot S_{1:d+1} \cdot \frac{p(p+1)(2p+1)}{d+1}. \end{aligned}$$

If d is even, $d = 2p$, then

$$\begin{aligned} \mathbb{E}[\Delta] &\leq \frac{1}{3} \cdot S_{1:d+1} \cdot \frac{p(p+1)(2p+1)}{2p+1} \\ &= \frac{1}{3} \cdot S_{1:d+1} \cdot p(p+1). \end{aligned}$$

If d is odd, $d = 2p + 1$, then

$$\begin{aligned}
\mathbb{E}[\Delta] &\leq \frac{1}{3} \cdot S_{1:d+1} \cdot \frac{p(p+1)(d)}{d+1} \\
&\leq \frac{1}{3} \cdot S_{1:d+1} \cdot \frac{p(p+1)(d+1)}{d+1} \\
&= \frac{1}{3} \cdot S_{1:d+1} \cdot p(p+1).
\end{aligned}$$

In summary, for any $u \in K$, its expected contribution to total persistence is,

$$\begin{aligned}
\mathbb{E}[\Delta] &\leq \frac{1}{3} \cdot S_{1:d+1} \cdot \lfloor \frac{d}{2} \rfloor (\lfloor \frac{d}{2} \rfloor + 1) \\
&\leq \frac{1}{3} \cdot S_{1:d_{min}+1} \cdot \lfloor \frac{d_{max}}{2} \rfloor (\lfloor \frac{d_{max}}{2} \rfloor + 1) \\
&= C_1.
\end{aligned}$$

□

It is interesting to note that, when K is a triangulation of \mathbb{S}^1 , where $d_{min} = d_{max} = 2$, $C_0 \approx C_1$. In other words, the bound in Theorem 3.4.3 is tight.

3.5 Change in Total Persistence

In this section, we prove Theorem 3.5.1.

Theorem 3.5.1 (Change in Total Persistence). *Let K be a triangulation of a general topological space with n vertices. Let $f, g : V \rightarrow \mathbb{R}$ be functions defined on V , such that for each vertex $u_i \in V$, $g(u) = f(u) + \varepsilon_i$, where $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n\} \stackrel{i.i.d.}{\sim} N(0, \sigma^2)$. We then obtain by linear extensions PL functions $f, g : |K| \rightarrow \mathbb{R}$. Then*

$$\mathbb{E}[\text{Pers}(g)] - \mathbb{E}[\text{Pers}(f)] \leq C \cdot n,$$

where d_{max} is the maximum degree of vertices in K , and $C = \sqrt{\frac{2}{\pi}} \sigma d_{max}$.

Proof. For a vertex $u \in K$, define Δ as its contribution to the change in total persistence as its function value changes from $f(u)$ to $g(u)$. The main step is to prove that $E[\Delta] \leq C$, where $C = \sqrt{\frac{2}{\pi}}\sigma d_{max}$. Since there are n vertices, then $E[\text{Pers}(g)] - E[\text{Pers}(f)] \leq C \cdot n$.

To compute the change in total persistence from f to g , suppose that we change the function $f : K \rightarrow \mathbb{R}$ in multiple steps:

$$f = f_0 \rightarrow f_1 \rightarrow \dots \rightarrow f_i \rightarrow \dots \rightarrow f_n = g,$$

where $f_i : K \rightarrow \mathbb{R}$ is defined by $f_i(u_j) = g(u_j)$ for $1 \leq j \leq i$, and $f_i(u_j) = f(u_j)$ for $i + 1 \leq j \leq n$. Therefore, according to Lemma 3.4.2, at each step i , when the function changes from f_{i-1} to f_i , the change in total persistence is

$$\text{Pers}(f_i) - \text{Pers}(f_{i-1}) \leq d_{max}|g(u_i) - f(u_i)|.$$

Then

$$\begin{aligned} & E[\text{Pers}(f_i)] - E[\text{Pers}(f_{i-1})] \\ & \leq d_{max}E[|g(u_i) - f(u_i)|] \\ & = d_{max}\sqrt{\frac{2}{\pi}}\sigma. \end{aligned}$$

Sum over all vertices,

$$\begin{aligned} & E[\text{Pers}(g)] - E[\text{Pers}(f)] \\ & = \sum_{i=0}^n E[\text{Pers}(f_i)] - E[\text{Pers}(f_{i-1})] \\ & \leq (d_{max}\sqrt{\frac{2}{\pi}}\sigma)n \\ & = Cn. \end{aligned}$$

□

Comparison. Now we’ve proved both Theorem 3.4.3 and Theorem 3.5.1 that are related to the total persistence in expectation. Set $\sigma = 1$ and $f = 0$ in the latter. We are curious about which theorem gives a tighter bound. In fact, this depends on the choice of d_{min} and d_{max} . For example, for a triangulation of \mathbb{S}^2 where $d_{min} = 3$ and $d_{max} = 6$, $C_1 = 4S_{1:4} \approx 4 \times 0.73237 = 2.92948$. Meanwhile, $C \approx 4.78731$. That is, $C_1 < C$. On the other hand, for a triangulation of a topological space where $d_{min} = 3$ and $d_{max} = 12$, then $C_1 = 14S_{1:4} \approx 14 \times 0.73237 = 10.25318$, while $C \approx 9.57461$. That is, $C < C_1$.

3.6 Discussion

There is much to discuss here. We focus on a few topics:

1. We notice that when running the extended persistence algorithm on a function defined on a triangulation of a general topological space, there are certain “simplicity” assumption for the underline space. Namely, we assume that in the persistence diagram, only points off the diagonal represent topological features. However, things become complicated for the triangulation of the projective plane where a simplex can pair with itself. This leads to a point on the diagonal in the persistence diagram which represents an important topological feature, even though its persistence is 0. We will need to extend the definition of extended persistence to capture such topological features.
2. The pairing probabilities given in Theorem 3.3.1 involve vertices from a triangulation of \mathbb{S}^1 . Are there similar results for vertices from a triangulation of \mathbb{S}^2 ?
3. We assume throughout this chapter that function values are sampled i.i.d. from a Normal distribution. Theorem 3.4.1 shows that the total persistence in expectation is linear in the number of vertices in the triangulation. Are there other sampling models that give sub-linear relations? What if we assume local dependencies, for example,

suppose the function values are m -dependent, identically distributed? Deriving the upper bound on total persistence is much harder and remains an open problem.

4. Theorem 3.5.1 gives an upper bound on the expected change in total persistence for functions with Gaussian noise defined on triangulations of general topological spaces. We can apply the theorem to estimate the total persistence in expectation for functions without noise. Can we prove tighter bounds?
5. Theorem 3.4.3 gives an upper bound on the total persistence in expectation. Suppose we know the expected multiplicities of vertices in a triangulation, it may be possible to derive a lower bound.
6. Under a similar setting from Theorem 3.5.1, can we compute a notion of “average persistence diagram” based on g ?

Chapter 4

Elevation

After studying the total persistence on some general triangulation, we now talk about triangulated surfaces in \mathbb{R}^3 and the volume they bound. Specifically, we use persistence to study features of triangulated protein surfaces. By features, we mean the protrusions and cavities on the surface of the protein which are relevant to forming complexes with other proteins during rigid-body docking.

4.1 Introduction

Motivation. The elevation function on a smoothly embedded 2-manifold in \mathbb{R}^3 reflects the multiscale topography of cavities and protrusions as local maxima. Introduced by Agarwal et al. [6], the function has been useful in identifying coarse docking configurations for protein pairs. The approach identifies protrusions (knobs) and cavities (wells) on the two surfaces and matches them up. This idea goes back to Connolly [42] who used a function that maps each point of the protein surface to the fraction of a fixed-radius sphere centered at the point that lies outside the protein volume. As shown by Cazals et al. [25], this function resembles the mean curvature at the point in the limit, when the radius approaches zero. The fixed radius makes a choice of the scale the function reflects.

In contrast to Connolly’s function, the elevation function is scale independent and marks small as well as large protrusions of varying shape and direction. Its construction is based on the persistence structure of the 2-parameter family of height functions, as explained in the next section. The task at hand is then the computation of all local maxima for two proteins and the use of the type, size, and location of the marked topographic features to identify promising positions for interaction. The experimental study in [112] shows that this approach is effective in finding initial positions that can then be refined by local optimization. The computationally most expensive step in this study is the determination of the elevation maxima. Using the algorithm in [6], the running time for a triangulated 2-manifold with m edges is proportional to $m^5 \log_2 m$. Since typical proteins give rise to surfaces with hundreds of thousands of edges, the quintic dependence on m limits the practical deployment of the method. Our goal is to compute local maxima faster in practice.

Results. In this chapter, we transport the concept of elevation function from the smooth to the piecewise linear category, and study its application in practice. Our main contributions are,

- We give a new algorithm for finding all local maxima. While its worse-case running time is the same as the algorithm used in [6], its performance is roughly ten-thousand times faster for triangulated surfaces approximating smooth surfaces that we typically find in practice.
- We cast light on this improvement by relating the running time to the total absolute Gaussian curvature of the 2-manifold and, to a lesser extent, to the number of vertices in the approximating triangulation.

All our experiments use molecular skin surfaces [49] as the triangulation. They are characterized by having dihedral angles at edges that are close to half the full angle. Since

we incorporate the surface complexity in terms of total absolute Gaussian curvature into the analysis of the algorithm, it is worth mentioning that there is a large literature on the notion of curvatures for triangulated surfaces. For example, several differential operator estimates for mean curvature, Gaussian curvature and principle curvature are derived for triangulated 2-manifolds in [77]. We refer to [12, 39, 80] for details.

4.2 Preliminaries

In this section, we introduce the geometric and topological background need to understand the elevation function and our algorithm for computing its local maxima. We begin with the mathematically cleaner smooth case, which we use as the guiding intuition in our subsequent treatment of the computationally more useful piecewise linear case.

The Smooth Case

We begin with a brief introduction of Morse functions and persistent homology, then use these concepts to define the elevation function. Finally, we discuss the Gaussian curvature of the 2-manifold.

Morse functions. The class of smooth, real-valued functions is a challenging object that simplifies considerably if we add genericity as a requirement. Since we will need to measure distance along the manifold, we assume \mathbb{M} is a 2-manifold with Riemannian metric defined on it. Letting $f : \mathbb{M} \rightarrow \mathbb{R}$ be a smooth function on the 2-manifold, a point $x \in \mathbb{M}$ is *critical* if the derivative at x equals zero. The value of f at a critical point is a *critical value*. All other points are *regular points* and all other values are *regular values* of f . A critical point is *non-degenerate* if the **Hessian**, that is, the matrix of second partial derivatives at the point is invertible. In the 2-dimensional case, we have a 2-by-2 Hessian, and if it is non-degenerate, then the matrix has two non-zero eigenvalues, $\lambda_1 \neq \lambda_2$. Define the *index* of the corresponding non-degenerate critical point as the number of negative

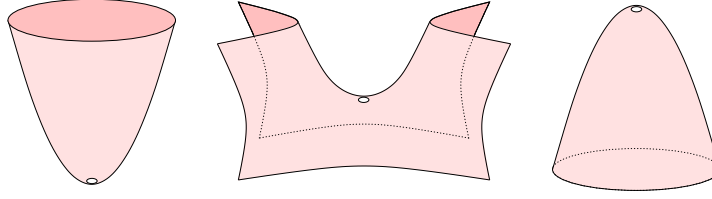


FIGURE 4.1: From left to right: a minimum, a saddle, and a maximum of the vertical height function.

eigenvalues. A non-degenerate critical point of index 0 is a *minimum*, of index 1 is a *saddle*, and of index 2 is a *maximum*, see Figure 4.1. Finally, f is a *Morse function* if all its critical points are non-degenerate and its values at the critical points are distinct.

Given a value $a \in \mathbb{R}$, the corresponding *sublevel set* consists of all points with value at most a ; that is, $\mathbb{M}_a = f^{-1}(-\infty, a]$. Sweeping the manifold in the direction of increasing function value, we get a 1-parameter family of sublevel sets. The topology of the sublevel set changes precisely when the sweep passes through a critical point. Let $t_1 < t_2 < \dots < t_n$ be the ordered sequence of critical values and $-\infty = s_0 < s_1 < \dots < s_n = \infty$ a sequence of interleaved values, that is, $s_i < t_{i+1} < s_{i+1}$, for all i . By assumption of f being Morse, we get from the sublevel set at s_i to the one at s_{i+1} by passing exactly one non-degenerate critical point. The change can be characterized in terms of the dimension of the handle we attach to go from \mathbb{M}_{s_i} to $\mathbb{M}_{s_{i+1}}$. For index 0, we add a 0-handle, that is, an isolated point which we then thicken to a disk. For index 1, we add a 1-handle, that is an interval attached to the boundary of the sublevel set at its endpoints which we then thicken to a strip. Finally, for index 2, we add a 2-handle, that is, a disk attached to the boundary of the sublevel set along its boundary circle.

Persistent homology. Looking at the homology groups [82] of the sequence of sublevel sets, we use the concept of persistence to measure the lengths of the intervals along which homology classes exist [51]. Since sublevel sets between two contiguous critical values

are homologically indistinguishable, we may consider the finite sequence

$$\emptyset = \mathbb{M}_0 \subseteq \mathbb{M}_1 \subseteq \dots \subseteq \mathbb{M}_n = \mathbb{M},$$

where we simplify notation by setting $\mathbb{M}_i = \mathbb{M}_{s_i}$. Fixing a dimension p ($p \geq 0$), each sublevel set has a p -th homology group and the sequence is connected from left to right by homomorphisms induced by inclusion, which we denote as $f_p^{i,j} : H_p(\mathbb{M}_i) \rightarrow H_p(\mathbb{M}_j)$. We have a *birth* at \mathbb{M}_i if the map $f_p^{i-1,i}$ is not surjective, and we have a *death* at \mathbb{M}_j if the map $f_p^{j-1,j}$ is not injective. Furthermore, the death at \mathbb{M}_j corresponds to the birth at \mathbb{M}_i if there is homology class γ in $H_p(\mathbb{M}_i)$ that is not in the image of $f_p^{i-1,i}$, its image in $H_p(\mathbb{M}_{j-1})$ is still not in the image of $f_p^{i-1,j-1}$, but its image in $H_p(\mathbb{M}_j)$ is in the image of $f_p^{i-1,j}$. We call $f(t_j) - f(t_i)$ the *persistence* of this birth-death pair. As explained in [34], this method gives a pairing between births and deaths that has many interesting properties. Each death corresponds to a unique birth but not every birth corresponds to a death. Missing the death is sometimes a problem because we can not get a measure for the critical point giving birth, like we can for all other critical points. This is especially true for the definition of the elevation function for which we need measurements of all critical points. To remedy this shortcoming, we extend the sequence of homology groups for extended persistence as described in [35]. Writing $\mathbb{M}^a = f^{-1}[a, \infty)$ for the *superlevel set* of a , we go up with absolute homology groups of sublevel sets, as before, and we come back down with relative homology groups,

$$\begin{aligned} 0 = H_p(\mathbb{M}_0) &\rightarrow H_p(\mathbb{M}_1) \rightarrow \dots \rightarrow H_p(\mathbb{M}_n) \\ &\rightarrow H_p(\mathbb{M}, \mathbb{M}^n) \rightarrow \dots \rightarrow H_p(\mathbb{M}, \mathbb{M}^0) = 0, \end{aligned}$$

where we simplify notation by setting $\mathbb{M}^i = \mathbb{M}^{s_i}$, $\mathbb{M}^0 = \mathbb{M}$ and $\mathbb{M}^n = \emptyset$. We call this the *extended filtration* and the resulting birth-death pairing the *extended persistence* of the function. Now every birth corresponds to a death. In fact, we have two events at every critical point, one going up and one coming down, but duality implies that we just get

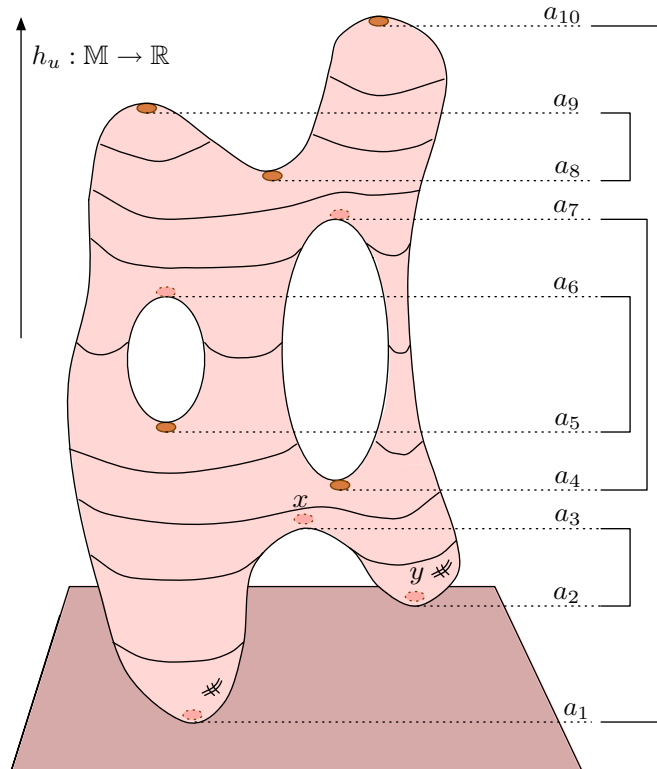


FIGURE 4.2: Example of an elevation function defined in the vertical height direction. The pairing of critical points are shown on the right.

each pair twice, see [35]. As a consequence of duality, the birth-death pairs we get for the negative function, $-f$, are the same. This turns out to be important in the definition of the elevation function.

For 2-manifolds, there is a more elementary way to introduce extended persistence using the Reeb graph of the function. Instead of giving details, we refer to [6] and we mention that this approach leads to a fast algorithm. It consists of constructing the Reeb graph in a sweep [40] followed by deconstructing it in another sweep using cutting and linking trees [6, 55].

Elevation. To define elevation, we assume the 2-manifold \mathbb{M} is smoothly embedded in \mathbb{R}^3 . For a direction $u \in \mathbb{S}^2$, we consider the **height function** $h_u : \mathbb{M} \rightarrow \mathbb{R}$ defined by $h_u(x) = \langle x, u \rangle$. Generically, h_u is a Morse function, but for some directions u it is not,

either because a critical point is degenerate or because two or more critical points map to the same height value. Considering the entire sphere of directions, we get a 2-parameter family of height functions.

For each $u \in \mathbb{S}^2$, we pair up births with deaths using the extended sequence of homology groups defined by the sublevel and the superlevel sets of h_u . In the Morse function case, each birth-death pair identifies two critical points, x and y , one giving birth and the other giving death, and we define the **elevation** at these two points as their persistence or, equivalently, the absolute height difference in the direction u , $E(x) = E(y) = |h_u(x) - h_u(y)|$. Each point of \mathbb{M} is critical in two directions, u and $-u$, and is thus assigned two values, the absolute height difference to the paired critical point in the two directions. Since $h_{-u} = -h_u$, the paired point is the same, so we get a unique value at every point. This is the **elevation function** of the 2-manifold, $E : \mathbb{M} \rightarrow \mathbb{R}$. See Figure 4.2 for an example.

To get a feeling for this function, we consider a protrusion (a mountain) of the 2-manifold. To measure the height of the mountain, we measure from the top down, to the first saddle that separates it from an even higher mountain. We can do this in various directions, so we do it to maximize the height. This might be in a direction along which the first saddle is ambiguous. Perhaps there are three such saddles at the same height value in this direction, similar to the third type in Figure 4.3 in which we have a degenerate monkey saddle with the same height difference to three minima. In this direction, we have two violations of genericity required for Morse functions, because there are three critical points with the same height value. Indeed, local maxima of E tend to arise along non-generic directions. An exception is the 1-legged maximum defined by only two critical points (with one leg between them). Besides this case, we have 2-legged maxima defined by three critical points, and 3- and 4-legged maxima defined by four critical points each; see Figure 4.3.

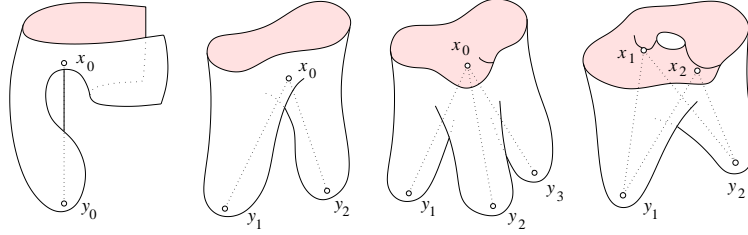


FIGURE 4.3: The four generic types of local maxima of the elevation function. From left to right: the 1-, 2-, 3- and 4-legged maximum.

Curvature. We will later discover that the running time of our algorithm for finding all local maxima relates to the total absolute curvature of the surface. We introduce this concept using the *Gauss map*, $N : \mathbb{M} \rightarrow \mathbb{S}^2$, defined by mapping a point x of \mathbb{M} to the outer unit normal, $N(x)$, at x . Assuming \mathbb{M} is smoothly embedded in \mathbb{R}^3 , the Gauss map is continuous and surjective but not necessarily injective. Indeed, the preimage of $u \in \mathbb{S}^2$ consists of all critical points of h_u with outer normal u , as opposed to $-u$. The multiplicity of N at u and $-u$ together is thus the number of critical points of h_u . We will see shortly that the total coverage of \mathbb{S}^2 is exactly the total absolute Gaussian curvature of \mathbb{M} .

Letting x be a point of \mathbb{M} and $r > 0$ a radius, we define the *absolute Gaussian curvature* at x by taking the limit of a fraction of areas, $g(x) = \lim_{r \rightarrow 0} \frac{\text{Area}(N(A_r))}{\text{Area}(A_r)}$, where A_r is the neighborhood of points at distance at most r from x on \mathbb{M} . The *total absolute Gaussian curvature* is the integral of the local quantity, $G(\mathbb{M}) = \int_{x \in \mathbb{M}} g(x) dx$. It should be clear that $G(\mathbb{M})$ is the area of the total coverage of \mathbb{S}^2 , taking multiplicity into account. For a given direction, the multiplicity is $|N^{-1}(u)|$. Hence, $G(\mathbb{M}) = \int_{u \in \mathbb{S}^2} |N^{-1}(u)| du$. Writing c_{avg} for the average number of critical points of the height functions, we thus have the total absolute Gaussian curvature equal to one half times the area of the sphere times that average, $G(\mathbb{M}) = 2\pi c_{\text{avg}}$. This integral geometry formula for the curvature will come handy in the analysis of our algorithm. For more information on the integral geometry formulation of curvature see Santaló [103].

The PL Case

We do all computations on a piecewise linear approximation of the smooth 2-manifold. To transport the smooth concepts to the PL category, we think of the PL surface as being approximated by a smooth surface. Tightening the approximation, we get a series and take the limit. This is the general intuition we have in the background guiding the formulation of definitions in the PL case.

Triangulated surfaces. A *triangulation* of a 2-manifold \mathbb{M} is a **simplicial complex**, K , whose underlying space is homeomorphic to \mathbb{M} : $|K| \approx \mathbb{M}$. It consists of vertices, edges, and triangles. To put K into \mathbb{R}^3 , it suffices to map each vertex to a point; the edges and triangles are the convex hulls (of the images) of their vertices. This is a *geometric realization* if the triangles meet in shared edges and vertices but not otherwise. We call the result a *triangulated surface*, implicitly assuming that it is geometrically realized in \mathbb{R}^3 . Here we recall some definitions that have appeared in Chapter 3. The *star* of a vertex is the set of simplices that contain it, and the *link* consists of all faces of simplices in the star that do not belong to the star:

$$\text{St } v_i = \{\sigma \in K \mid v_i \in \sigma\};$$

$$\text{Lk } v_i = \{\tau \subseteq \sigma \in \text{St } v_i \mid \tau \not\subseteq \text{St } v_i\}.$$

A PL function $f : |K| \rightarrow \mathbb{R}$ is determined by its values at the vertices. Assuming $f(v_i) \neq f(v_j)$ whenever $i \neq j$, we define the *lower link* as the subset of simplices in the link where f is smaller than at the vertex; and the *lower star* as the subset of simplicies for which v_i is the vertex with the maximum function value:

$$\text{Lk}_{-}v_i = \{\sigma \in \text{Lk } v_i \mid x \in \sigma \Rightarrow f(x) < f(v_i)\},$$

$$\text{St}_{-}v_i = \{\sigma \in \text{St } v_i \mid x \in \sigma \Rightarrow f(x) \leq f(v_i)\}.$$

Note that there are partial simplicies where $f(x) > f(v_i)$ at some points, and $f(x) < f(v_i)$ at others. Finally, v_i is *regular vertex* if its lower link is contractible, and *critical vertex*, otherwise. Since K triangulates a 2-manifold, every link is a circle and the only contractible closed subsets are points and closed paths.

The lower link of a regular vertex is thus a single vertex or a path connecting two vertices. A *minimum* is characterized by $\text{Lk}_-v_i = \emptyset$ and a *maximum* by $\text{Lk}_-v_i = \text{Lk } v_i$. In the remaining case, the lower link consists of $k + 1 \geq 2$ paths and we call v_i a *k-fold saddle*, or a *simple saddle* if $k = 1$ (see examples in Figure 3.1, Chapter 3). Recall that we can then classify the vertices using the reduced Betti numbers of their lower link (Table 3.1, Chapter 3).

In contrast to the smooth case, it is not possible to turn a k -fold into a simple saddle by a small perturbation. We therefore treat them directly, without reduction to simple cases. As an example, consider the Euler-Poincaré Theorem which relates the topology of the 2-manifold with the critical point structure of its functions. Denote the *index* of a simple critical point by $\text{index}(v_i)$,

$$\text{index}(v_i) = \begin{cases} 0 & \text{if } v_i \text{ is a minimum;} \\ 1 & \text{if } v_i \text{ is a simple saddle;} \\ 2 & \text{if } v_i \text{ is a maximum.} \end{cases}$$

Notice that this PL version of index is one more than the dimension of the unique non-zero reduced Betti number of the lower link. It is also consistent with the corresponding definition in the smooth case. Assuming K is connected, it is characterized by its *genus* and we have

$$2 - 2 \cdot \text{genus} = n - m + l = \sum_i (-1)^{\text{index}(v_i)},$$

where n, m, l are the number of vertices, edges, triangles in K and a k -fold saddle is represented by k simple saddles in the sum.

Critical regions. Another significant complication we encounter in the PL case is that a vertex is generally critical for an entire region of directions. Letting $h_u : |K| \rightarrow \mathbb{R}$ be the height function defined by $h_u(x) = \langle x, u \rangle$, the *critical region* of a vertex is the closure of the set of directions along which v_i is critical,

$$R_i = \text{cl} \{u \in \mathbb{S}^2 \mid v_i \text{ is critical point of } h_u\}.$$

We construct it from the closed polygonal curve defined by the star of v_i . Specifically, we map each triangle in the star to its outer normal direction, a point on \mathbb{S}^2 , and we connect the directions of two neighboring triangles by the shorter of the two connecting great-circle arcs. This gives a closed polygonal curve, π_i , which may or may not have self-intersections. To cope with self-intersections, we orient π_i and define the *winding number* of a direction $u \in \mathbb{S}^2$ not on the curve as the number of times the curve goes around the directed line defined by u . Viewed along u , we count a counterclockwise turn as $+1$ and a clockwise turn as -1 . Taking the sum we get the winding number, which we denote as $w(u, \pi_i)$. For a detailed study of the Gauss map for polyhedral surfaces, refer to [9]. The winding number of u relates to the type of the vertex in the height function defined by u . Specifically, if v_i is regular then the winding number of u is 0, if v_i is a simple critical point then the winding number is $(-1)^{\text{index}(v_i)}$, and if v_i is a k -fold saddle then the winding number is $-k$. Examples are shown in Figure 4.4 where a monkey saddle has the winding number -2 .

Curvature. Thinking of a vertex as a tiny region in an approximating smooth surface, we define its *Gaussian curvature* as the area of its critical region weighted by the winding number. More useful in this chapter is its *absolute Gaussian curvature* defined as the area weighted by the absolute winding number, $g(v_i) = \int_{u \in \mathbb{S}^2} |w(u, \pi_i)| du$. The *total absolute Gaussian curvature* is then the sum over all vertices, $G(K) = \sum_i g(v_i)$. Equivalently, it is the area of the sphere times half the average number of critical vertices, taking multiplic-

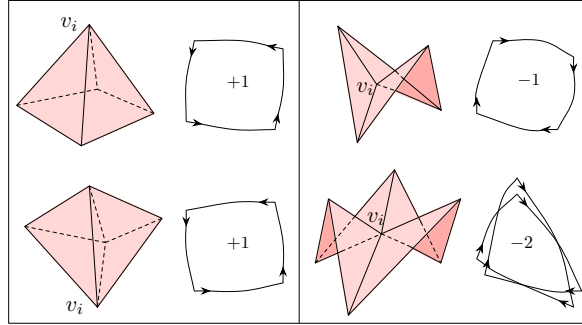


FIGURE 4.4: Left: for a direction u with winding number $+1$ the corresponding vertex appears either as a maximum or a minimum. Right: for winding number -1 we have a simple saddle and for -2 we have a 2-fold or monkey saddle for the height function defined by the corresponding direction.

ities into account, as usual. The average is taken over all height functions, and we count half the critical vertices because v_i is critical for $u \in \mathbb{S}^2$ as well as $-u \in \mathbb{S}^2$.

4.3 Computation

In this section, we describe how we compute the elevation maxima for a given triangulated surface in \mathbb{R}^3 . The algorithm is straightforward and the new insight relative to prior work is in the analysis, relating the running time with the total absolute Gaussian curvature of the surface.

Types and filters. Recall that there are four types of elevation maxima for a generic smooth surface, as illustrated in Figure 4.3. We have the same four cases for a generic triangulated surface K in \mathbb{R}^3 . Each maximum is given by a set of two, three, or four points. We consider the case in which all these points are vertices of K . The cases in which some of the points in V lie on edges of K are similar. Let V be a set of vertices. A necessary requirement for V to define an elevation maximum is that its vertices are critical for a common height function. More specifically, we need them critical in a particular direction that is determined by V . This direction, $u_V = (y - x)/\|y - x\|$, is slightly different for each type.

1-legged case, $V = \{x, y\}$. Here, u_V is the direction defined by the two points.

2-legged case, $V = \{x, y_1, y_2\}$. Letting y be the orthogonal projection of x onto the line passing through y_1 and y_2 , $u_V = (y - x)/\|y - x\|$, if y lies between y_1 and y_2 .

3-legged case, $V = \{x, y_1, y_2, y_3\}$. Letting y be the orthogonal projection of x onto the plane passing through y_1, y_2, y_3 , $u_V = (y - x)/\|y - x\|$, if y lies in the triangle they span.

4-legged case, $V = \{x_1, x_2, y_1, y_2\}$. Letting x and y be the feet of the shortest line segment connecting the line passing through x_1 and x_2 with the line passing through y_1 and y_2 , $u_V = (y - x)/\|y - x\|$, if x lies between x_1 and x_2 and y lies between y_1 and y_2 .

With the definition of u_V , now we are ready to introduce one of the two filters, the projection filter, which is useful in eliminating the collection of point sets with empty common critical regions.

PROJECTION FILTER. The direction u_V defined by the points in V is defined and belongs to the common intersection of critical regions, $u_V \in \bigcap_{v_i \in V} R_i$.

Notice that the four cases discussed so far are “vertex-only”, that is, each V contains only points located on the vertices. There are other cases where V contains points that lie on edges of K , which potentially form elevation local maxima. These mixed sets can be generated by substituting edges connecting adjacent vertices in vertex-only sets. For example, the four vertices of a 4-legged case may give rise to a mixed set containing two vertices and one edge, specifying a 2-legged case, or a set of two edges, specifying a 1-legged case.

Note that the non-empty intersection of the critical regions is a necessary but not a sufficient condition for the set V to pass the Projection Filter. In turn, passing the Projection Filter is a necessary but not sufficient condition for the direction u_V to be an elevation maximum. For that, the set needs to satisfy another condition. To describe it, we write x_0 for x .

PERSISTENCE FILTER. For each pair x_i and y_j in V , there is an arbitrarily small perturbation u of u_V such that x_i, y_j is a birth-death pair for the height function h_u .

Algorithm. We compute the elevation maxima in three steps, starting with 2-, 3-, 4-tuplets V whose points have pairwise overlapping critical regions. The next two steps narrow down the selection using first the Projection and second the Persistence Filter.

STEP 0. Compute the critical regions of the vertices of K . Letting the critical regions be the nodes of the intersection graph, R , we draw an arc if the two regions have a non-empty common intersection. For $k = 2, 3, 4$, let Q_k be the set of k -cliques, that is, the k -tuplets of nodes connected by all $\binom{k}{2}$ arcs. Let $S_0 = \bigcup_{k=2}^4 Q_k$.

STEP 1. Subject each pair, triplet, and quadruplet in S_0 to the Projection Filter and let $S_1 \subseteq S_0$ be the collection of sets in S_0 that passes the filter.

STEP 2. Subject each pair, triplet, and quadruplet in S_1 to the Persistence Filter and let $S_2 \subseteq S_1$ be the collection of sets in S_1 that passes the filter.

Steps 1 and 2 are the same as in [6], so we focus on the implementation of Step 0 in which we compute the 2-, 3-, 4-tuplets with pairwise intersecting critical regions.

Implementation. We decompose Step 0 into three smaller steps, constructing the critical regions, finding the intersecting pairs, and computing the cliques of size 2, 3, 4 in the intersection graph. Implementation is done with Perl, C and CGAL [2]. All computations are exact except estimating the area and the bounding box of a critical region.

STEP 0.1. Recall that each critical region, R_i , is given by a closed polygon with m_i edges on the sphere. Recall that m_i is the number of edges incident to vertex v_i in the triangulation. Those edges may intersect, and we take time $O(m_i^2)$ to construct the decomposition of the sphere [43], including winding numbers for all subregions. Reflecting R_i centrally through the origin in \mathbb{R}^3 , we get the region $-R_i$ of inward normals along which v_i is critical. Constructing all critical regions takes time proportional to $\sum_i m_i^2$.

STEP 0.2. We found experimentally that most critical regions are small and simple. This suggests we use a bounding volume approach to find the intersecting pairs. Specifically, we find an axis-parallel box B_i in \mathbb{R}^3 that encloses the region R_i on $\mathbb{S}^2 \subseteq \mathbb{R}^3$. We do this in two steps, first computing the smallest enclosing sphere of R_i and second the smallest axis-aligned box that contains the sphere. Assuming that R_i fits inside a hemisphere of \mathbb{S}^2 , the smallest enclosing sphere of its vertices also encloses R_i . To compensate for round-off errors, we increase the sphere slightly and compute the box B_i to enclose the enlarged sphere. Computing the smallest enclosing sphere of R_i takes randomized time $O(m_i)$, see [114]. Given the boxes B_i , we find the overlapping pairs using the segment-tree streaming algorithm as described in [116]. Writing b_i for the number of boxes that overlap B_i , we have a total of $b = \frac{1}{2} \sum_i b_i$ overlapping pairs. The streaming algorithm takes time proportional to $n \log_2^3 n + b$ to find them. For each pair of overlapping boxes, we check whether or not the critical region they enclose have a non-empty intersection. Standard computational geometry methods allow us to determine whether or not R_i and R_j intersect in time $O(m_{ij} \log m_{ij})$, where $m_{ij} = m_i^2 + m_j^2$ [43].

STEP 0.3. The result of Steps 0.1 and 0.2 is a graph R . Its n nodes are the critical regions, and its q arcs are the pairs of critical regions with non-empty overlap. Writing $q = \frac{1}{2} \sum_i q_i$, where q_i is the degree of the i -th node, we compute the cliques of

size 2, 3, 4 by checking all pairs and triplets of neighbors. Finding the cliques that include the i -th node, R_i , thus takes time $O(\binom{q_i}{1} + \binom{q_i}{2} + \binom{q_i}{3})$.

Analysis. The time for Step 0 is dominated by the requirement for Step 0.2, which is some constant times $T_{\text{new}} = \sum_i (\binom{q_i}{1} + \binom{q_i}{2} + \binom{q_i}{3})$. The time for Step 1 is some constant times $|S_0| \leq T_{\text{new}}$ and that for Step 2 is some constant times $T = |S_1|n \log_2 n$. This adds up to some constant times $T_{\text{new}} + T$, as compared to $T_{\text{old}} + T$ for the algorithm in [6], where $T_{\text{old}} = \binom{n}{2} + \binom{n}{3} + \binom{n}{4}$. Any improvement thus hinges on two properties, namely that T_{old} is significantly larger than T_{new} as well as T . We now show that the first property holds under grossly simplifying assumptions, and we provide evidence in the next section that both properties hold for data we encounter in practice. Here is the simplifying assumption we use:

CAP ASSUMPTION. The critical regions are spherical caps, all of the same size, and their centers are uniformly distributed on \mathbb{S}^2 .

Recall that the areas of the critical regions add up to the total absolute Gaussian curvature, $\sum_i \text{Area}(R_i) = G(K)$. This sum is also half the area of the sphere times the average number of critical points of the height functions, $G(K) = 2\pi c_{\text{avg}}$. It follows the area of a single critical region is $\text{Area}(R_i) = 2\pi c_{\text{avg}}/n$, and because the cap is smaller than the flat disk of the same radius, its radius squared is $\rho^2 > 2c_{\text{avg}}/n$. Two caps overlap if and only if the center of one is contained in the cap of radius 2ρ around the center of the other. The area of the enlarged cap is less than four times $\text{Area}(R_i)$. Hence the probability for a region R_j to overlap R_i is $\Pr(R_i \cap R_j \neq \emptyset) \leq 4\text{Area}(R_i)/4\pi = 2c_{\text{avg}}/n$. Since expectations are additive even if the events are not independent, the expected number of k -tuples of neighbors is $\text{Exp}[\binom{q_i}{k}] \leq \binom{n-1}{k} \text{Area}(R_i)^k / \pi^k \leq 2^k c_{\text{avg}}^k / k!$. Adding the expectations

for $k = 1, 2, 3$ and all $1 \leq i \leq n$ gives

$$\text{Exp}[T_{\text{new}}] \leq n \cdot (2c_{\text{avg}} + 2c_{\text{avg}}^2 + \frac{4}{3}c_{\text{avg}}^3).$$

Recall that $c_{\text{avg}} = G(K)/2\pi$. It follows the average number of k -tuplets of critical regions overlapping a given one depends on the shape of the smooth surface and not on the size of the approximating triangulated surface. Similarly, the time for Step 0 depends on the shape and otherwise only linearly on the number of vertices in the triangulation.

4.4 Experiments

In this section, we present the results of our computational experiments. Running our software on triangulated surfaces representing biomolecular structures, we gather statistics on critical regions, pairwise intersections, and elevation maxima. We use these statistics as evidence that the Cap Assumption is a reasonable approximation of the reality for our data and that the new algorithm runs about four orders of magnitude faster than the old one.

Input data. We use two types of triangulated surfaces approximating smooth models of biomolecular structures all listed in Table 4.1. The first type is the molecular skin which uses hyperboloid and concave sphere patches to blend between the spheres that represent the atoms of a molecule [49]. An algorithm that constructs an approximating triangulated surface with guaranteed bounds on two- and three-dimensional angles is described in [29] and software written by Ho-lun Cheng is available at [1]. For a representative of our data set, see Figure 4.5. The second type is the molecular surfaces generated by Chimera [92]. The MSMS algorithm used in Chimera [102] constructs a triangulation of the solvent excluded surfaces initially computed by Connolly [41]. We compute local maxima for the skin surfaces only as they tend to have finer triangulations with smaller percentage of non-simple critical vertices compared with the Chimera surfaces. We suspect that the

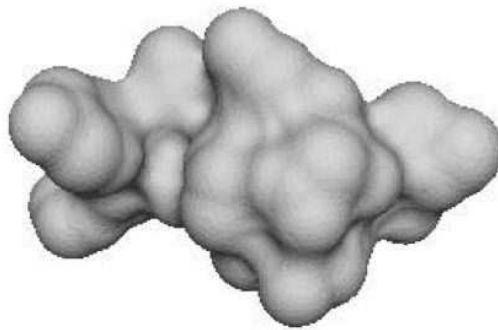


FIGURE 4.5: Representative of our data sets, a triangulated surface approximating a peptide within the 1BRS protein.

algorithm speed-up might be sensitive to the quality of the triangulation, specifically to large average size of the critical regions, which might give rise to larger set of candidates of local maxima.

id	name	n	m	l
0	1BRS-5to6	1,370	4,104	2,736
1	1CLU-DBG	3,149	9,441	6,294
2	1BRS-A-5to10	4,248	12,738	8,492
3	1BRS-A-30to40	6,114	18,336	12,224
4	1BRS-A-17to25	7,799	23,391	15,594
5	1BRS-A-5to10	836	2,502	1,668
6	1BRS-A-30to40	1,372	4,110	2,740
7	1BRS-A-17to25	1,595	4,119	3,186

TABLE 4.1: The triangulated surfaces used in our computational experiments together with their numbers of vertices, edges, and triangles. Top: five molecular skin surfaces. Bottom: those molecular Chimera surfaces.

Critical point statistics. For each data set, we estimate the minimum, average, and maximum number of critical points of the height functions, which we sample at one thousand directions chosen from \mathbb{S}^2 . The results are shown in Table 4.2, left. Comparing the estimated with the actual average, which we get using $c_{\text{avg}} = G(K)/2\pi = \sum_i \text{Area}(R_i)/2\pi$, we see that the error is small. For example, for data set 4, the estimated c_{avg} is 29.92 while the actual average is 29.94. Since all our skin triangulations approximate a smooth surface to about the same accuracy, for different surfaces, the average number of critical points

scales linearly with n . Indeed, c_{avg}/n is between 0.003 and 0.005 for all our skin data sets.

As mentioned earlier, each vertex of K is critical for a region of directions, in fact two antipodal regions. Most of these regions are simple, that is, defined by a polygon without self-intersections. As shown in Table 4.2 right, the percentage of non-simple polygons is indeed rather small. Besides checking for self-intersections, we measure the complexity of a critical region by counting the triangles we need to triangulate it on the sphere. The minimum, average, and maximum of this number are given in Table 4.2 middle.

id	c_{\min}	c_{avg}	c_{\max}	$\frac{c_{\text{avg}}}{n}$	r_{\min}	r_{avg}	r_{\max}	%
0	2	6.41	16	0.0047	2	3.99	8	12
1	2	13.50	44	0.0043	2	4.01	12	15
2	6	17.07	34	0.0040	2	4.01	10	17
3	10	25.14	46	0.0041	2	4.01	10	16
4	12	29.92	64	0.0038	2	4.01	10	20
5	6	16.01	32	0.0192	2	4.08	11	29
6	10	27.13	46	0.0198	2	4.13	15	30
7	14	31.02	54	0.0194	2	4.09	10	33

TABLE 4.2: Left: estimated minimum, average, and maximum of the number of critical points of the height functions. Middle: minimum, average, and maximum of the number of triangles needed to triangulate the critical regions. Right: percentage of non-simple critical regions. Top: five molecular skin surfaces. Bottom: those molecular Chimera surfaces.

Intersection statistics. The following statistics were collected for the finer molecular skin surfaces only. Recall that we compute the pairs of intersecting critical regions in two steps, first finding the intersections among the bounding boxes and second among the critical regions. Table 4.3, left, gives the statistics for both. Given a pair of intersecting

id	b_{\min}	b_{avg}	b_{\max}	$\frac{b_{\text{avg}}}{n}$	q_{\min}	q_{avg}	q_{\max}	$\frac{q_{\text{avg}}}{n}$
0	12	94	207	0.069	9	40	97	0.029
1	27	204	626	0.065	11	82	250	0.026
2	52	236	556	0.056	20	92	201	0.022
3	95	243	859	0.040	29	134	330	0.022
4	99	423	1,276	0.054	35	160	543	0.021

TABLE 4.3: Left: the minimum, average, and maximum number of boxes intersecting a given box; Right: the minimum, average, and maximum number of critical regions intersection a given critical region.(Data computed for the five molecular skin surfaces only.)

boxes, we test whether or not the corresponding critical regions intersect by checking the overlap among the triangles in their triangulations. The average number of triangle-triangle checks is consistently between 11 and 12, which justifies the use of this brute-force over a more sophisticated method.

Similar to the number of critical points, we expect that the average number of boxes intersecting a given box and the average number of critical regions intersecting a given critical region to scale linearly with n . Indeed, b_{avg}/n is between 0.04 and 0.07 and q_{avg}/n is between 0.02 and 0.03 for all our skin data sets. The latter is about six times the average number of critical points; compare this with the factor two we got under the Cap Assumption. The observed relation between these two quantities is only about three times as loose, which is reasonable considering that real data necessarily violates the Cap Assumption to some extent (due to irregular shapes and different orientations of the critical regions). The new algorithm starts with T_{new} tuplets. A back-of-the-envelope calculation suggests that T_{new} is roughly $n^{\binom{q_{\text{avg}}}{3}}$, which is roughly a factor of ten thousand smaller than $\binom{n}{4}$, independent of the value of n . We thus might expect the new algorithm to run about four orders of magnitude faster than the old one.

Running time. Recall that S_0 is the set of cliques of size 2, 3, or 4 in the intersection graph of the critical regions. The subset $S_1 \subseteq S_0$ contains all cliques that pass the Projection Filter, and the subset $S_2 \subseteq S_1$ contains all cliques that also pass the Persistence Filter. The sizes of the first two sets are given in Table 4.4 left.

id	$ S_0 /10^3$	$ S_1 $	$T_{\text{old}}/10^{10}$	$T_{\text{new}}/10^6$	$T/10^6$
0	1,608	2,373	15	24	33
1	32,119	20,521	410	508	749
2	43,572	17,175	1,356	720	882
3	198,023	56,797	5,820	3,327	4,368
4	433,116	94,300	15,411	7,354	9,508

TABLE 4.4: Left: the number of cliques before and after the Projection Filter and the Persistence Filter. Right: dominant terms in the running time of the old and the new algorithms. (Data computed for the five molecular surfaces only.)

Most relevant to the running time of the algorithms for computing elevation maxima is S_1 . Indeed, both the old and the new algorithm start with sets of 2-, 3-, and 4-tuplets that contain the cliques in S_0 and much more. As shown in Table 4.4 on the right, the overestimate by the old algorithm is about ten thousand times that of the new algorithm. Furthermore, in the new algorithm, the time for Step 0 and Steps 1 and 2 is fairly balanced. This implies a speed-up of about four orders of magnitude, which is consistent with back-of-the-envelope calculation mentioned above.

4.5 Conclusion and Discussion

The main result of this chapter is a new algorithm for computing all elevation maxima of a triangulated surface in \mathbb{R}^3 . We provide experimental evidence that for practical data, the new algorithm runs about four orders of magnitude faster than the old one. The improvement is achieved by making the running time dependent on the total absolute Gaussian curvature of the surface and to a lesser extent on the number of vertices in the approximating triangulation. There are several open problems as follows.

1. Now, the total absolute Gaussian curvature has different definitions for smooth and for piecewise linear surfaces. It appears that $G(K)$ approaches $G(\mathbb{M})$ as K is refined and forms a progressively more accurate approximation of the smooth surface \mathbb{M} . However, we do not have a proof and we do not know under what conditions this is true.
2. There is room for performance improvement, one promising direction is to parallelize the computations. It would also be interesting to sample the elevation maxima if this can be done faster than computing all. For example, is it possible to compute all elevation maxima larger than some threshold without spending the time to determine (and discard) the elevation maxima that do not exceed that threshold?

Chapter 5

Towards Stratification Learning through Homology Inference

A topological approach to stratification learning is developed for point cloud data drawn from a stratified space. The objective is, given the point cloud data, infer which points belong to the same strata. Topological conditions are given under which the point cloud can be used to infer properties of the stratified space. Kernel and cokernel persistent homology is used to state these conditions which characterize the local structure of points in the sample. A geometric intuition for the topological conditions is provided. We state finite sample bounds on the minimum number of points in the sample required to state with high probability which points belong to the same strata. We present an algorithm that computes which points belong to the same strata and prove the correctness of this algorithm. The algorithm is applied to simulated data.

5.1 Introduction

A basic problem in geometry, topology, and statistical inference that has received attention in the past is that of manifold learning: given a point cloud of data sampled from a man-

ifold in an ambient space \mathbb{R}^k , infer the underlying manifold. A limitation of the problem statement is that it does not apply to sets that are not manifolds. For example, we may consider the more general class of stratified spaces that can be decomposed into strata, which are manifolds of varying dimension, each of which fit together in some way uniformly inside the higher dimensional space. In this chapter, we study the following problem in stratification learning: given a point cloud sampled from a stratified space, which points belong to a common stratum?

Consistency in manifold learning has been characterized as homology inference: as the number of points in a point cloud goes to infinity, the inferred homology converges to the true homology of the underlying space. Results of this nature have been given for manifolds [88, 89] and a large class of compact subsets of Euclidean space [27]. Stronger results in homology inference for closed subsets of a metric space are given in [34].

Geometric approaches to stratification inference have been developed before. These include inference of a mixture of linear subspaces [73], mixture models for general stratified spaces [61], and generalized Principal Component Analysis (GPCA) [110] which was developed for dimension reduction for mixtures of manifolds.

The study of stratified spaces has been a focus of pure mathematics [59, 113]. Computational and algorithmic work on this topic has been developed in the study of intersection homology [58] and of persistence for intersection homology [19]. The problem of inference of local homology in a deterministic setting has been addressed in [20].

Results. In this chapter we propose an approach to stratification inference based on homology inference. The results in this chapter are,

- A topological characterization of two points belonging to the same stratum by assessing the local structure of the points through kernel and cokernel persistent homology;

- Topological conditions on the point sample under which the topological characterization holds – we call this topological inference;
- A geometric intuition of these topological conditions based on geometric quantities related to reach and the gradient of a distance function;
- Finite sample bounds for the minimum number of points in the sample required to state with high probability which points belong to the same strata;
- An algorithm that computes which points belong to the same strata and a proof of correctness of this algorithm.

5.2 Background

We review necessary background on persistence, homology and stratified spaces. The treatment here is mostly adapted from [26]. We first develop persistence modules that arise from maps between homology groups induced by inclusions of topological spaces. We then discuss stratifications and their connection to the local homology groups of a topological space. We assume basic knowledge of homology itself, referring the reader to [82] or [63] or [50] for a more computationally oriented treatment.

5.2.1 Persistence Modules

In [26], the authors define persistence modules over an arbitrary commutative ring R with unity. For simplicity, we restrict immediately to the case $R = \mathbb{Z}/2\mathbb{Z}$. Let A be some subset of \mathbb{R} . Then a *persistence module* \mathcal{F}_A is a family $\{F_\alpha\}_{\alpha \in A}$ of $\mathbb{Z}/2\mathbb{Z}$ -vector spaces, together with a family $\{f_\alpha^\beta : F_\alpha \rightarrow F_\beta\}_{\alpha \leq \beta \in A}$ of linear maps such that $\alpha \leq \beta \leq \gamma$ implies $f_\alpha^\gamma = f_\beta^\gamma \circ f_\alpha^\beta$. We will assume that the index set A is either \mathbb{R} or $\mathbb{R}_{\geq 0}$ and not explicitly state indices unless necessary.

A real number α is said to be a *regular value* of the persistence module \mathcal{F} if there

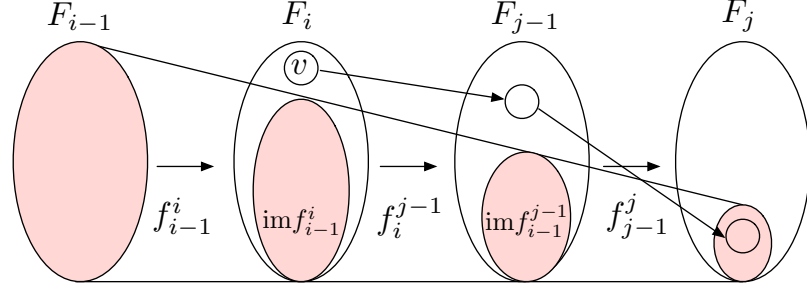


FIGURE 5.1: Persistence modules.

exists some $\epsilon > 0$ such that, for all $\delta < \epsilon$, the maps $f_{\alpha-\delta}^{\alpha+\delta}$ are all isomorphisms. Otherwise we say that α is a *critical value* of the persistence module; if $A = \mathbb{R}_{\geq 0}$, then $\alpha = 0$ will always be considered to be a critical value. We say that \mathcal{F} is *tame* if it has a finite number of critical values and if all the vector spaces F_α are of finite rank. If $\mathcal{F}_{\mathbb{R}_{\geq 0}}$ is tame it has a smallest non-zero critical value $\rho(\mathcal{F})$; we call this number the *feature size* of the persistence module.

Assume \mathcal{F} is tame and so we have a finite ordered list of critical values $0 = c_0 < c_1 < \dots < c_m$. We choose regular values $\{a_i\}_{i=0}^m$ such that $c_{i-1} < a_{i-1} < c_i < a_i$ for all $1 \leq i \leq m$, and we adopt the shorthand notation $F_i \equiv F_{a_i}$ and $f_i^j : F_i \rightarrow F_j$, for $0 \leq i \leq j \leq m$. A vector $v \in \mathcal{F}_i$ is said to be *born* at level i if $v \notin \text{im } f_{i-1}^i$, and such a vector *dies* at level j if $f_i^j(v) \in \text{im } f_{i-1}^j$ but $f_i^{j-1}(v) \notin \text{im } f_{i-1}^{j-1}$, here im stands for image. We then define $P^{i,j}$ to be the vector space of vectors that are born at level i and then subsequently die at level j , and $\beta^{i,j}$ denotes its rank. This is illustrated in Figure 5.1.

Persistence Diagrams

The information contained within a tame module \mathcal{F} can be compactly represented by a *persistence diagram*, $\text{Dgm}(\mathcal{F})$. This diagram is a multi-set of points in the *extended plane*. It contains $\beta^{i,j}$ copies of the points (c_i, c_j) , as well as infinitely many copies of each point along the major diagonal $y = x$. In Figure 5.3 the persistence diagrams for a curve and a point cloud sampled from it are displayed.

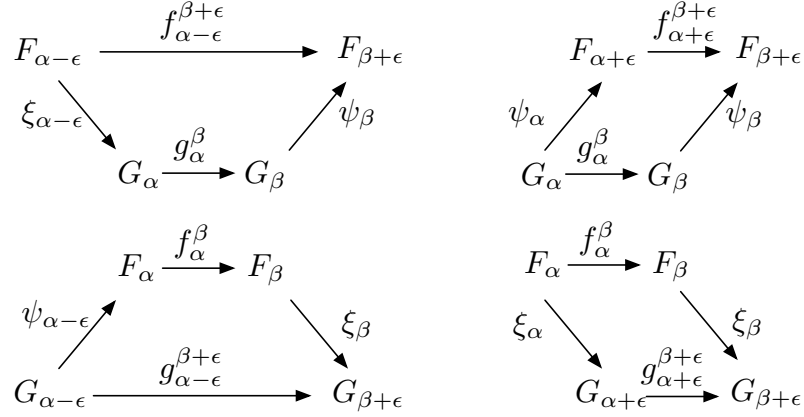


FIGURE 5.2: Commuting diagrams for strongly interleaving persistence modules.

A distance metric between persistence diagrams can be defined that has a stability property, see Theorem 5.2.1 – if two persistence modules are “close” then the corresponding persistence diagrams are “close”.

For any two points $u = (x, y)$ and $u' = (x', y')$ in the extended plane, we define $\|u - u'\|_{\infty} = \max\{|x - x'|, |y - y'|\}$. We define the *bottleneck distance* between any two persistence diagrams D and D' to be:

$$d_B(D, D') = \inf_{\Gamma: D \rightarrow D'} \sup_{u \in D} \|u - \Gamma(u)\|_{\infty},$$

where Γ ranges over all bijections from D to D' . Under certain conditions which we now describe, persistence diagrams will be stable under the bottleneck distance.

Two persistence modules \mathcal{F} and \mathcal{G} are said to be *strongly ϵ -interleaved* if, for some positive ϵ , there exist two families $\{\xi_{\alpha} : F_{\alpha} \rightarrow G_{\alpha+\epsilon}\}_{\alpha}$ and $\{\psi_{\alpha} : G_{\alpha} \rightarrow F_{\alpha+\epsilon}\}_{\alpha}$ of linear maps which commute with the module maps $\{f_{\alpha}^{\beta}\}$ and $\{g_{\alpha}^{\beta}\}$ in the appropriate manner. More precisely, we require, for all $\alpha \leq \beta$, $f_{\alpha-\epsilon}^{\beta+\epsilon} = \psi_{\beta} \circ g_{\alpha}^{\beta} \circ \xi_{\alpha-\epsilon}$ and $\psi_{\beta} \circ g_{\alpha}^{\beta} = f_{\alpha+\epsilon}^{\beta+\epsilon} \circ \psi_{\alpha}$, as well as the two other equations obtained by exchanging the roles of f and g and ξ and ψ . This is shown in Figure 5.2.

We can now state the diagram stability result ([26], Theorem 4.4), that we will need later in this chapter.

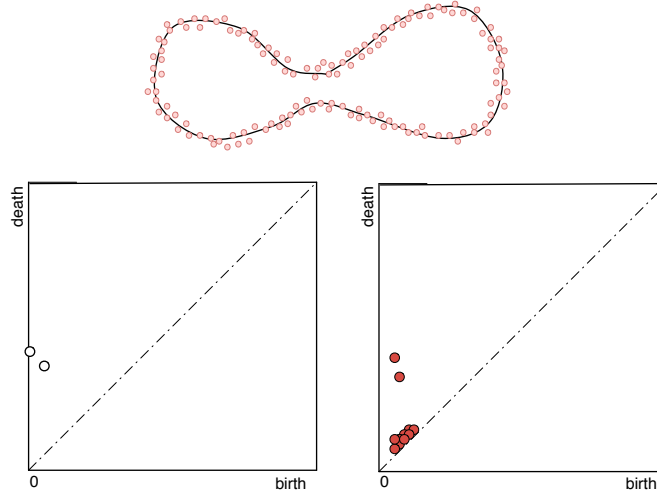


FIGURE 5.3: Illustration of a point cloud and its persistence diagram. Top: \mathbb{X} is the curve embedded as shown in the plane and U is the point cloud. Bottom left: the persistence diagram of $\text{Dgm}_1(d_{\mathbb{X}})$; Bottom right: the persistence diagram of $\text{Dgm}_1(d_U)$.

Theorem 5.2.1 (Diagram Stability Theorem). *Let \mathcal{F} and \mathcal{G} be tame persistence modules and $\epsilon > 0$. If \mathcal{F} and \mathcal{G} are strongly ϵ -interleaved, then*

$$d_B(\text{Dgm}(\mathcal{F}), \text{Dgm}(\mathcal{G})) \leq \epsilon.$$

When we wish to compute the persistence diagram associated to a module \mathcal{F} , it is often convenient to substitute another module \mathcal{G} , usually one defined in terms of simplicial complexes or other computable objects. The following theorem ([50], p.159) gives a condition under which this is possible.

Theorem 5.2.2 (Persistence Equivalence Theorem). *Given two persistence modules \mathcal{F} and \mathcal{G} , suppose there exist for each α isomorphisms $F_\alpha \cong G_\alpha$ which commute with the module maps, then $\text{Dgm}(\mathcal{F}) = \text{Dgm}(\mathcal{G})$.*

That is, if all the vertical maps are isomorphisms and all squares commute in the fol-

lowing diagram, then $\text{Dgm}(\mathcal{F}) = \text{Dgm}(\mathcal{G})$.

$$\begin{array}{ccc} \dots & \rightarrow & F_\alpha \rightarrow F_\beta \rightarrow \dots \\ & & \uparrow \cong \quad \uparrow \cong \\ \dots & \rightarrow & G_\alpha \rightarrow G_\beta \rightarrow \dots \end{array}$$

(Co)Kernel Modules

Suppose now that we have two persistence modules \mathcal{F} and \mathcal{G} along with a family of maps $\{\phi_\alpha : F_\alpha \rightarrow G_\alpha\}$ which commute with the module maps – for every pair $\alpha \leq \beta$, we have $g_\alpha^\beta \circ \phi_\alpha = \phi_\beta \circ f_\alpha^\beta$. That is,

$$\begin{array}{ccc} \dots & \rightarrow & F_\alpha \xrightarrow{f_\alpha^\beta} F_\beta \rightarrow \dots \\ & & \downarrow \phi_\alpha \quad \downarrow \phi_\beta \\ \dots & \rightarrow & G_\alpha \xrightarrow{g_\alpha^\beta} G_\beta \rightarrow \dots \end{array}$$

For each $\alpha \leq \beta$, the restriction of f_α^β to $\ker \phi_\alpha$ maps into $\ker \phi_\beta$, giving rise to a new **kernel** persistence module, with persistence diagram denoted by $\text{Dgm}(\ker \phi)$. That is,

$$\dots \rightarrow \ker \phi_\alpha \rightarrow \ker \phi_\beta \rightarrow \dots$$

Similarly, we obtain a **cokernel** persistence module, with diagram $\text{Dgm}(\text{cok } \phi)$.

5.2.2 Homology

Our main examples of persistence modules all come from homology groups, either absolute or relative, and the various maps between them. Homology persistence modules can arise from families of topological spaces $\{\mathbb{X}_\alpha\}$, along with inclusions $\mathbb{X}_\alpha \hookrightarrow \mathbb{X}_\beta$ for all $\alpha \leq \beta$. Whenever we have such a family, the inclusions induce maps $H_j(\mathbb{X}_\alpha) \rightarrow H_j(\mathbb{X}_\beta)$, for each homological dimension $j \geq 0$, and hence we have persistence modules for each j . Defining $H(\mathbb{X}_\alpha) = \bigoplus_j H_j(\mathbb{X}_\alpha)$ and taking direct sums of maps in the obvious way, will also give one large direct-sum persistence module $\{H(\mathbb{X}_\alpha)\}$.

Distance Functions

Here, the families of topological spaces will be produced by the sublevel sets of distance functions. Given a topological space \mathbb{X} embedded in some Euclidean space \mathbb{R}^k , we define $d_{\mathbb{X}}$ as the **distance function** which maps each point in the ambient space to the distance from its closest point in \mathbb{X} . More formally, for each $y \in \mathbb{R}^k$, $d_{\mathbb{X}}(y) = \inf_{x \in \mathbb{X}} \text{dist}(x, y)$. We let \mathbb{X}_{α} denote the sublevel set $d_{\mathbb{X}}^{-1}[0, \alpha]$; each sublevel set should be thought of a thickening of \mathbb{X} within the ambient space. As the thickening parameter increases, the thickenings include one another, giving rise to the persistence module $\{\mathbb{H}(\mathbb{X}_{\alpha})\}_{\alpha \in \mathbb{R}_{\geq 0}}$; we denote the persistence diagram of this module by $\text{Dgm}(d_{\mathbb{X}})$ and use $\text{Dgm}_j(d_{\mathbb{X}})$ for the diagrams of the individual modules for each homological dimension j .

In Figure 5.3, we see an example of such an \mathbb{X} embedded in the plane, along with the persistence diagram $\text{Dgm}_1(d_{\mathbb{X}})$. We also have the persistence diagram $\text{Dgm}_1(d_U)$, where U is a dense point sample of \mathbb{X} . Note that the two diagrams are quite close in bottleneck distance. Indeed, we will always have $d_B(\text{Dgm}(d_{\mathbb{X}}), \text{Dgm}(d_U)) \leq \epsilon$, where $\epsilon = d_H(\mathbb{X}, U)$ is the Hausdorff distance between the space and its sample; this follows from Theorem 5.2.1.

Persistence modules of relative homology groups arise from families of pairs of spaces, as the next example shows. Referring to the left part of Figure 5.4, we let \mathbb{X} be the space drawn in solid lines and B the closed ball whose boundary is drawn as a dotted circle. By restricting $d_{\mathbb{X}}$ to B and also to ∂B , we produce pairs of sublevel sets $(\mathbb{X}_{\alpha} \cap B, \mathbb{X}_{\alpha} \cap \partial B)$. Using the maps induced by the inclusions of pairs, we obtain the persistence module $\{\mathbb{H}(\mathbb{X}_{\alpha} \cap B, \mathbb{X}_{\alpha} \cap \partial B)\}_{\alpha \in \mathbb{R}_{\geq 0}}$ of relative homology groups. The persistence diagram, for homological dimension 1, appears on the right half of Figure 5.4.

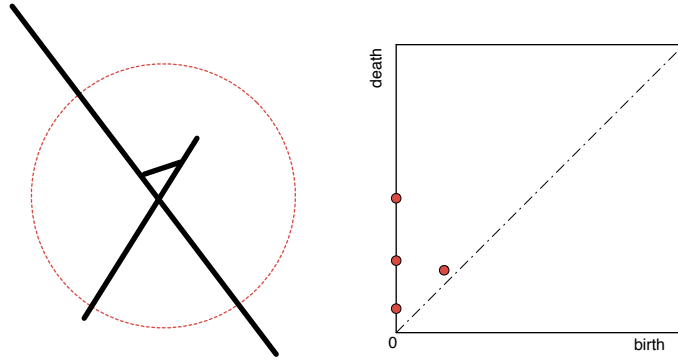


FIGURE 5.4: Left: The space \mathbb{X} is in solid line and the closed ball B has dotted boundary. Right: the persistence diagram for the module $\{H_1(\mathbb{X}_\alpha \cap B, \mathbb{X}_\alpha \cap \partial B)\}$.

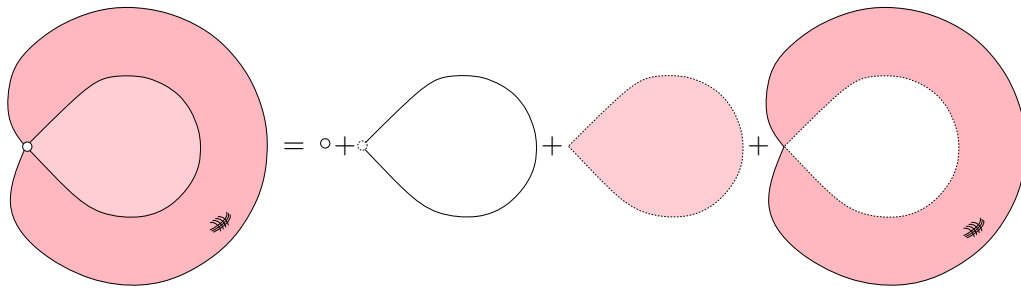


FIGURE 5.5: Example of the stratification of a pinched torus with a spanning disc stretched across the hole.

5.2.3 Stratified Spaces

We assume that we have a topological space \mathbb{X} embedded in some Euclidean space \mathbb{R}^k . A d -dimensional *stratification* of \mathbb{X} is a decreasing sequence of closed subspaces

$$\mathbb{X} = \mathbb{X}_d \supseteq \mathbb{X}_{d-1} \supseteq \dots \supseteq \mathbb{X}_0 \supseteq \mathbb{X}_{-1} = \emptyset,$$

such that for each i , the i -dimensional *stratum* $\mathbb{S}_i = \mathbb{X}_i - \mathbb{X}_{i-1}$ is a (possibly empty) i -manifold. The connected components of \mathbb{S}_i are called i -dimensional *pieces*. This is illustrated in Figure 5.5, where the space \mathbb{X} is a pinched torus with a spanning disc stretched across the hole.

One usually also imposes a requirement to ensure that the various pieces fit together uniformly. There are a number of different ways this can be done (see [65] for an extensive

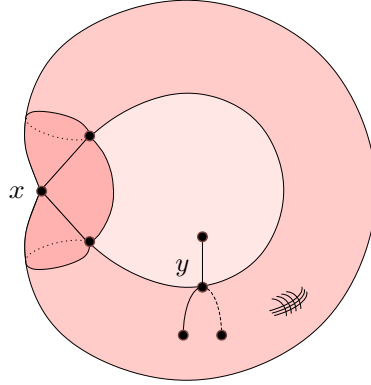


FIGURE 5.6: A 2-dimensional stratified space. The cones $c(L_x)$ and $c(L_y)$, where x and y are respectively in the 0-stratum and the 1-stratum, are highlighted.

survey). For example, one might assume that for each $x \in \mathbb{S}_i$, there exists a small enough neighborhood $N(x) \subseteq \mathbb{X}$ and a $(d - i - 1)$ -dimensional stratified space L_x such that $N(x)$ is stratum-presevering homeomorphic to the product of an i -ball and the cone on L_x ; one can then show that the space L_x depends only on the particular piece containing x . This definition is illustrated in Figure 5.6, again the space \mathbb{X} is a pinched torus with a spanning disc stretched across the hole.

Since the topology on \mathbb{X} is that inherited from the ambient space, this neighborhood $N(x)$ will take the form $\mathbb{X} \cap B_r(x)$, where $B_r(x)$ is a small enough ball around x in the ambient space.

Local Homology and Homology Stratifications

Recall ([82]) that the local homology groups of a space \mathbb{X} at a point $x \in \mathbb{X}$ are the groups $H_i(\mathbb{X}, \mathbb{X} - x)$ in each homological dimension i . If \mathbb{X} happens to be a d -manifold, or if x is simply a point in the top-dimensional stratum of a d -dimensional stratification, then these groups are rank one in dimension d and trivial in all other dimensions. On the other hand, the local homology groups for lower-stratum points can be more interesting; for example if x is the crossing point in Figure 5.7, then $H_1(\mathbb{X}, \mathbb{X} - x)$ has rank three.

If x and y are close enough points in a particular piece of the same stratum, then there

is a natural isomorphism between their local homology groups $H(\mathbb{X}, \mathbb{X} - x) \cong H(\mathbb{X}, \mathbb{X} - y)$, which can be understood in the following manner. Taking a small enough radii r and using excision, we see that the two local homology groups in question are in fact just $H(\mathbb{X} \cap B_r(x), \mathbb{X} \cap \partial B_r(x))$ and $H(\mathbb{X} \cap B_r(y), \mathbb{X} \cap \partial B_r(y))$. Both of these groups will then map, via intersection of chains, isomorphically into the group $H(\mathbb{X} \cap B_r(x) \cap B_r(y), \partial(B_r(x) \cap B_r(y)))$, and the isomorphism above is then derived from these two maps. See the points in Figure 5.7 for an illustration of this idea.

In [101], the authors define the concept of a homology stratification of a space \mathbb{X} . Briefly, they require a decomposition of \mathbb{X} into pieces such that the locally homology groups are locally constant across each piece; more precisely, that the maps discussed above be isomorphisms for each pair of close enough points in each piece. This is interesting because in computations we will not be able to distinguish anything finer.

Whitney Stratification and Stratified Morse Theory

Let S_i and S_j be two pieces of a stratification of \mathbb{X} . A stratification is called a *Whitney stratification* if for every pair of S_i and S_j with $S_i \subset \text{cl } S_j$, the following Whitney conditions A and B hold [59, 76]. Suppose that two sequences of points $\{y_k\} \in S_i$ and $\{x_k\} \in S_j$ converges to $y \in S_i$. Suppose that the secant lines $\overline{x_k y_k}$ converge to some limiting line l , and that the tangent spaces at x_k to S_j , $T_{x_k} S_j$, converge to some limiting space T (called *generalized tangent space at y*). Then,

A. $T_y S_i \subset T$

B. $l \subset T$

Condition B implies condition A [74]. Any triangulable stratified space is Whitney and any Whitney stratified space can be triangulated [59].

Let \mathbb{X} be a d -dimensional Whitney stratified space embedded in some smooth manifold \mathbb{M} . Let $\bar{f} : \mathbb{M} \rightarrow \mathbb{R}$ be a smooth function. The restriction f of \bar{f} to \mathbb{X} is *critical* at a point

$x \in \mathbb{X}$ iff it is critical when restricted to the particular manifold piece which contains x [19]. A *critical value* of f is its value at a critical point. f is a *Stratified Morse function* iff [19],

1. f is a Morse function when restricted to each manifold piece.
2. All critical values of f are distinct.
3. The differential of f at a critical point $x \in S_i$ does not annihilate any generalized tangent space to x other than $T_x S_i$.

We now state without proof the first fundamental theorem of Stratified Morse Theory. This will be useful in proving Lemma 5.4.2.

Theorem 5.2.3 (First Fundamental Theorem of Stratified Morse Theory). *Let \mathbb{X} be a Whitney stratified space and f a real-valued Stratified Morse function on it. Suppose the interval $[a, b]$ contains no critical values of f . Then the sublevel set $f^{-1}(-\infty, b]$ deformation retracts, preserving strata, onto $f^{-1}(-\infty, a]$. In particular, the sets have the same homology.*

5.3 Topological Inference Theorem

A result of the relationship between local homology groups and stratification is that any stratification of a topological space will also be a homology stratification. The converse is unfortunately false. However, we can build a useful analytical tool based on the contrapositive: given two points in a point cloud we can state based on their local homology groups, and the maps between them that the two points should not be placed in the same piece of any stratification. To do this, we first adapt the definition of these local homology maps into a more multi-scale and robust framework. This involves the introduction of a radius parameter r and defining a notion of local equivalence at different scales, values of r . There are two main results in this section. The first is an equivalence relation between

two points $x, y \in \mathbb{X}$. The second uses this equivalence relation to stratify a point cloud U sampled from \mathbb{X} .

5.3.1 Local Equivalence

We assume that we are given some topological space \mathbb{X} embedded in some Euclidean space in \mathbb{R}^k . For each radius $r \geq 0$, and for each pair of points $p, q \in \mathbb{X}$, we define the following homology map:

$$\phi^{\mathbb{X}}(p, q, r) : H(\mathbb{X} \cap B_r(p), \mathbb{X} \cap \partial B_r(p)) \rightarrow H(\mathbb{X} \cap B_r(p) \cap B_r(q), \mathbb{X} \cap \partial(B_r(p) \cap B_r(q))). \quad (5.1)$$

Intuitively, this map can be understood as taking a chain, throwing away the parts that lie outside the smaller range, and then modding out the new boundary. Alternatively, one may think of it as being induced by a combination of inclusion and excision. A formal definition is given in Appendix A.

Using these maps, we impose the following equivalence relation on \mathbb{X} .

Definition 5.3.1 (Local equivalence). *Two points x and y are said to have equivalent local structure at radius r , denoted $x \sim_r y$, iff there exists a chain of points $x = x_0, x_1, \dots, x_m = y$ from \mathbb{X} such that, for each $1 \leq i \leq m$, the maps $\phi^{\mathbb{X}}(x_{i-1}, x_i, r)$ and $\phi^{\mathbb{X}}(x_i, x_{i-1}, r)$ are both isomorphisms.*

In other words, x and y have the same local structure at this radius iff they can be connected by a chain of points which are pairwise close enough and whose local homology groups at radius r map into each other via intersection. Different choices of r will of course lead to different equivalence classes. For example, consider the space \mathbb{X} drawn in the plane as shown in the left half of Figure 5.7. At the radius drawn, point z is equivalent to the cross point and is not equivalent to either the point x or y . On the other hand, a smaller choice of radius would result in x, y, z belonging to the same class.

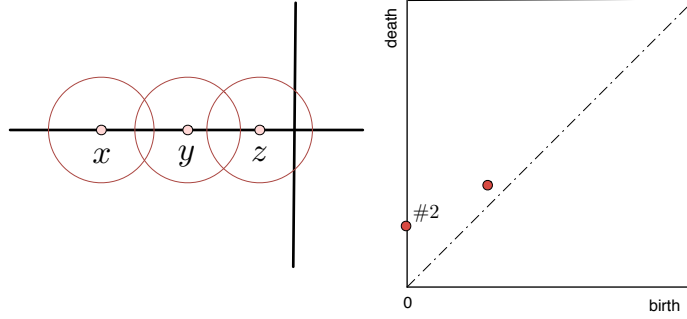


FIGURE 5.7: Left: $x \sim_r y$, $y \approx_r z$. Right: the 1-dim persistence diagram, for the kernel of the map going from the z ball into its intersection with the y ball. A number, i.e., #2, labeling a point in the persistence diagram indicates its multiplicity.

(Co)Kernel Persistence

We define a multi-scale version of $\phi^{\mathbb{X}}(p, q, r)$ by thickening the space \mathbb{X} . Let $d_{\mathbb{X}} : \mathbb{R}^k \rightarrow \mathbb{R}$ denote the function which maps each point in the ambient space to the distance to its closest point on \mathbb{X} . For each $\alpha \geq 0$, we define $\mathbb{X}_{\alpha} = d_{\mathbb{X}}^{-1}[0, \alpha]$. Fixing p and q , we write $\phi_{\alpha}^{\mathbb{X}}$ for $\phi^{\mathbb{X}}(p, q, r)$, using the above notation in map 5.1 substituting \mathbb{X}_{α} for \mathbb{X} . Note of course that $\phi^{\mathbb{X}} = \phi_0^{\mathbb{X}}$.

In the remainder of this chapter, we adapt the following shorthand,

$$B_p^{\mathbb{X}}(\alpha) = \mathbb{X}_{\alpha} \cap B_r(p),$$

$$\partial B_p^{\mathbb{X}}(\alpha) = \mathbb{X}_{\alpha} \cap \partial B_r(p),$$

$$B_{pq}^{\mathbb{X}}(\alpha) = \mathbb{X}_{\alpha} \cap B_r(p) \cap B_r(q),$$

$$\partial B_{pq}^{\mathbb{X}}(\alpha) = \mathbb{X}_{\alpha} \cap \partial(B_r(p) \cap B_r(q)).$$

When $\alpha = 0$, we write $B_p^{\mathbb{X}} = B_p^{\mathbb{X}}(0)$, $B_{pq}^{\mathbb{X}} = B_{pq}^{\mathbb{X}}(0)$.

For any pair of non-negative real values $\alpha < \beta$ the inclusion $\mathbb{X}_{\alpha} \hookrightarrow \mathbb{X}_{\beta}$ gives rise to

the following commutative diagram:

$$\begin{array}{ccc}
\mathrm{H}(B_p^{\mathbb{X}}(\alpha), \partial B_p^{\mathbb{X}}(\alpha)) & \xrightarrow{\phi_\alpha^{\mathbb{X}}} & \mathrm{H}(B_{pq}^{\mathbb{X}}(\alpha), \partial B_{pq}^{\mathbb{X}}(\alpha)) \\
\downarrow & & \downarrow \\
\mathrm{H}(B_p^{\mathbb{X}}(\beta), \partial B_p^{\mathbb{X}}(\beta)) & \xrightarrow{\phi_\beta^{\mathbb{X}}} & \mathrm{H}(B_{pq}^{\mathbb{X}}(\beta), \partial B_{pq}^{\mathbb{X}}(\beta))
\end{array} \tag{5.2}$$

Hence there are maps $\ker \phi_\alpha^{\mathbb{X}} \rightarrow \ker \phi_\beta^{\mathbb{X}}$ and $\mathrm{cok} \phi_\alpha^{\mathbb{X}} \rightarrow \mathrm{cok} \phi_\beta^{\mathbb{X}}$. Allowing α to increase from 0 to ∞ gives rise to two persistence modules, $\{\ker \phi_\alpha^{\mathbb{X}}\}$ and $\{\mathrm{cok} \phi_\alpha^{\mathbb{X}}\}$, with diagrams $\mathrm{Dgm}(\ker \phi^{\mathbb{X}})$ and $\mathrm{Dgm}(\mathrm{cok} \phi^{\mathbb{X}})$. Recall that a homomorphism is an isomorphism iff its kernel and cokernel are both zero. In our context, the map $\phi^{\mathbb{X}}$ is an isomorphism iff neither $\mathrm{Dgm}(\ker \phi^{\mathbb{X}})$ nor $\mathrm{Dgm}(\mathrm{cok} \phi^{\mathbb{X}})$ contain any points on the y -axis above 0.

Example. As shown in the left part of Figure 5.7, x , y and z are points sampled from the underling space (which is a cross). Let $p = z$ and $q = y$. The right part of the figure displays $\mathrm{Dgm}_1(\ker \phi^{\mathbb{X}})$, which we now explain in some detail. The group $\mathrm{H}_1(B_p^{\mathbb{X}}, \partial B_p^{\mathbb{X}})$ has rank three; as a possible basis we might take the three classes represented by the horizontal line across the ball, the vertical line across the ball, and the two short segments defining the northeast-facing right angle. Under the intersection map $\phi^{\mathbb{X}} = \phi_0^{\mathbb{X}}$, the first of these classes maps to the generator of $\mathrm{H}_1(B_{pq}^{\mathbb{X}}, \partial B_{pq}^{\mathbb{X}})$, while the other two map to zero. Hence $\ker \phi_0^{\mathbb{X}}$ has rank two. Both classes in this kernel eventually die, one at the α value which fills in the northeast corner of the larger ball, and the other at the α value which fills in the entire right half, which happens at the same time. At this latter value, the map $\phi_\alpha^{\mathbb{X}}$ is an isomorphism and it remains so until the intersection of the two balls fills in completely. This gives birth to a new kernel class which subsequently dies when the larger ball finally fills in. The diagram $\mathrm{Dgm}_1(\ker \phi^{\mathbb{X}})$ thus contains three points; the leftmost two show that the map $\phi^{\mathbb{X}}$ is not an isomorphism.

5.3.2 Inference Theorem

Given a point cloud U sampled from \mathbb{X} consider the following question: for a radius r , how can we infer whether or not any given pair of points in U has the same local structure at this radius? In this subsection, we prove a theorem which describes the circumstances under which we can make the above inference. This requires using of U to infer whether or not the maps $\phi^{\mathbb{X}}(p, q, r)$ are isomorphisms. The basic idea is that if U is a dense enough sample of \mathbb{X} , then the (co)kernel diagrams defined by U will be good approximations of the diagrams defined by \mathbb{X} .

(Co)Kernel Stability

Fix p, q , and r , and remember that $\phi^{\mathbb{X}} = \phi^{\mathbb{X}}(p, q, r)$. If the y -axes of $\text{Dgm}(\ker \phi^{\mathbb{X}})$ and $\text{Dgm}(\text{cok } \phi^{\mathbb{X}})$ are empty above 0, then $\phi^{\mathbb{X}}$ is an isomorphism. We are not given the space \mathbb{X} but are given a point cloud U which we use to approximate these diagrams. For each $\alpha \geq 0$, we let $U_\alpha = d_U^{-1}[0, \alpha]$. We consider $\phi_\alpha^U = \phi_\alpha^U(p, q, r)$, defined by setting $\mathbb{X} = U_\alpha$ in map 5.1. Running α from 0 to ∞ , we obtain two more persistence modules, $\{\ker \phi_\alpha^U\}$ and $\{\text{cok } \phi_\alpha^U\}$, with diagrams $\text{Dgm}(\ker \phi^U)$ and $\text{Dgm}(\text{cok } \phi^U)$.

If U is a dense enough sample of \mathbb{X} , then the (co)kernel diagrams defined by U will be good approximations of the diagrams defined by \mathbb{X} . More precisely, we have the following corollary of Theorem 5.2.1:

Theorem 5.3.2 ((Co)Kernel Diagram Stability). *For the map ϕ^U the following stability properties hold:*

$$\begin{aligned} d_B(\text{Dgm}(\ker \phi^U), \text{Dgm}(\ker \phi^{\mathbb{X}})) &\leq d_H(U, \mathbb{X}), \\ d_B(\text{Dgm}(\text{cok } \phi^U), \text{Dgm}(\text{cok } \phi^{\mathbb{X}})) &\leq d_H(U, \mathbb{X}). \end{aligned}$$

Proof. We prove the first inequality; the proof of the second is identical. Put $\epsilon = d_H(U, \mathbb{X})$. Then, for each $\alpha \geq 0$, the inclusions $U_\alpha \hookrightarrow \mathbb{X}_{\alpha+\epsilon}$ and $\mathbb{X}_\alpha \hookrightarrow U_{\alpha+\epsilon}$ induce maps $\ker \phi_\alpha^U \rightarrow$

$\ker \phi_{\alpha+\epsilon}^{\mathbb{X}}$ and $\ker \phi_{\alpha}^{\mathbb{X}} \rightarrow \ker \phi_{\alpha+\epsilon}^U$. These maps clearly commute with the module maps in the needed way, and hence we have the required ϵ -interleaving and can thus appeal to Theorem 5.2.1. \square

Main Inference Result

Suppose that the point cloud U is a good representation of the space \mathbb{X} . Specifically the Hausdorff distance between U and \mathbb{X} is small, $d_H(U, \mathbb{X}) \leq \epsilon$, we call U an ϵ -approximation of \mathbb{X} . We compute diagrams $\text{Dgm}(\ker \phi^U)$ and $\text{Dgm}(\text{cok } \phi^U)$ from U , we provide an algorithm for this in Section 5.6. Consider two points $p, q \in U$ and a fixed radius r , our objective is to state conditions under which we can determine whether $p \sim_r q$ based on $\text{Dgm}(\ker \phi^U)$ and $\text{Dgm}(\text{cok } \phi^U)$. From the definition of local equivalence, Definition 5.3.1, this translates into checking whether the map $\phi^{\mathbb{X}}$ is an isomorphism.

We first define some relevant quantities and then we state the first main theorem of this section. Given any persistence diagram \mathcal{D} and two positive real numbers $a < b$, we let $\mathcal{D}(a, b)$ denote the multi-set of points of \mathcal{D} which lie in the portion of the extended plane which lies above $y = b$ and to the left of $x = a$; note that these points correspond to classes which are born no later than a and die no earlier than b . For a fixed p, q, r , we consider two spaces: $B_p^{\mathbb{X}}$ and $\partial B_{pq}^{\mathbb{X}}$. For each space, we imagine thickening it and noting the first time at which some absolute or relative homological change occurs. We then define $\rho(p, q, r)$ to be the minimum of these two values. More precisely, there are two persistence modules: $\{\text{H}(B_p^{\mathbb{X}}(\alpha), \partial B_p^{\mathbb{X}})\}$ and $\{\text{H}(B_{pq}^{\mathbb{X}}(\alpha), \partial B_{pq}^{\mathbb{X}})\}$. We let $\sigma(p, r)$ and $\sigma(p, q, r)$ denote their respective feature sizes and then set $\rho(p, q, r)$ to their minimum, $\rho(p, q, r) = \min\{\sigma(p, r), \sigma(p, q, r)\}$, and $\rho(q, p, r)$ is defined similarly, $\rho(q, p, r) = \min\{\sigma(q, r), \sigma(p, q, r)\}$.

We now state the main theorem of this section. This theorem states that we can use U to decide whether or not p and q have the same local structure at radius r , as long as $\rho(p, q, r)$ and $\rho(q, p, r)$ are both large enough relative to the sampling density.

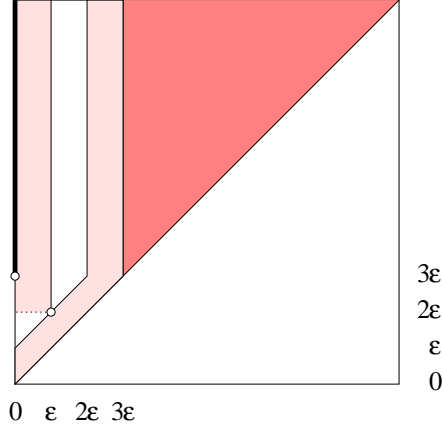


FIGURE 5.8: The points in the \mathbb{X} -diagrams lie either along the solid black line or in the darkly shaded region. Adding the lightly shaded regions, we get the region of possible points in the U -diagrams.

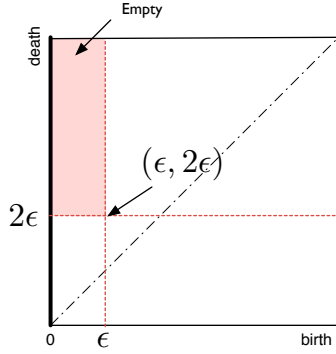


FIGURE 5.9: Empty rectangle in the ker/cok persistence diagrams.

Theorem 5.3.3 (Topological Inference Theorem). *Given a point sample U from \mathbb{X} with $d_H(U, \mathbb{X}) \leq \epsilon$, if for two points $p, q \in U$, $\rho \geq 3\epsilon$, then $\phi^{\mathbb{X}}(p, q, r)$ is an isomorphism iff*

$$\text{Dgm}(\ker \phi^U)(\epsilon, 2\epsilon) \cup \text{Dgm}(\text{cok } \phi^U)(\epsilon, 2\epsilon) = \emptyset.$$

If both $\phi^{\mathbb{X}}(p, q, r)$ and $\phi^{\mathbb{X}}(q, p, r)$ are isomorphisms, then $p \sim_r q$.

This is illustrated in Figure 5.9.

Proof. Assume $\rho \geq 3\epsilon$. To simplify exposition, we will refer to points in $\text{Dgm}(\ker \phi^{\mathbb{X}}) \cup \text{Dgm}(\text{cok } \phi^{\mathbb{X}})$ and $\text{Dgm}(\ker \phi^U) \cup \text{Dgm}(\text{cok } \phi^U)$ as \mathbb{X} -points and U -points, respectively.

Whenever $0 < \alpha < \beta < 3\epsilon < \rho$, the two vertical maps in diagram (5.2) will by definition both be isomorphisms. Hence the maps $\ker \phi_\alpha^{\mathbb{X}} \rightarrow \ker \phi_\beta^{\mathbb{X}}$ and $\text{cok} \phi_\alpha^{\mathbb{X}} \rightarrow \text{cok} \phi_\beta^{\mathbb{X}}$ must also be isomorphisms. As α increases from 0 to ∞ , any element of the (co)kernel of $\phi^{\mathbb{X}}$ must live until at least 3ϵ , and any (co)kernel class which is born after 0 must in fact be born after 3ϵ . In other words, any \mathbb{X} -point must lie either to the right of the line $x = 3\epsilon$ or along the y -axis and above the point $(0, 3\epsilon)$; see Figure 5.8. Recall that $\phi^{\mathbb{X}}$ is an isomorphism iff $\ker \phi^{\mathbb{X}} = 0 = \text{cok} \phi^{\mathbb{X}}$. Thus $\phi^{\mathbb{X}}$ is an isomorphism iff the black line in Figure 5.8 contains no \mathbb{X} -points.

On the other hand, Theorem 5.3.2 requires that every U -point must lie within ϵ of an \mathbb{X} -point. That is, all U -points are contained within the two lightly shaded regions drawn in Figure 5.8. Since the right such region is more than ϵ away from the thick black line, there will be a U -point in the left region iff there is an \mathbb{X} -point on the thick black line. But the U -points within the left region are exactly the members of $\text{Dgm}(\ker \phi^U)(\epsilon, 2\epsilon) \cup \text{Dgm}(\text{cok} \phi^U)(\epsilon, 2\epsilon)$. Isomorphism of $\phi^{\mathbb{X}}(p, q, r)$ and $\phi^{\mathbb{X}}(q, p, r)$ implies the local equivalence condition. \square

Recall that $p \sim_r q$ iff the maps $\phi^{\mathbb{X}}(p, q, r)$ and $\phi^{\mathbb{X}}(q, p, r)$ are both isomorphisms. The theorem thus says that we can use U to decide whether or not p and q have the same local structure at radius r , as long as $\rho(p, q, r)$ and $\rho(q, p, r)$ are both large enough relative to the sampling density.

The following corollary clusters points according to strata and is a direct result of the above theorem.

Corollary 5.3.4 (Strata clustering). *Assume a point sample U from \mathbb{X} with $d_H(U, \mathbb{X}) \leq \epsilon$ and $\rho \geq 3\epsilon$ for all pairs of points $p, q \in U$. Each cluster C_i is the transitive closure of points $p, q \in U$ with the relation $p \sim_r q$. Points in the same cluster belong to the same stratum at resolution r .*

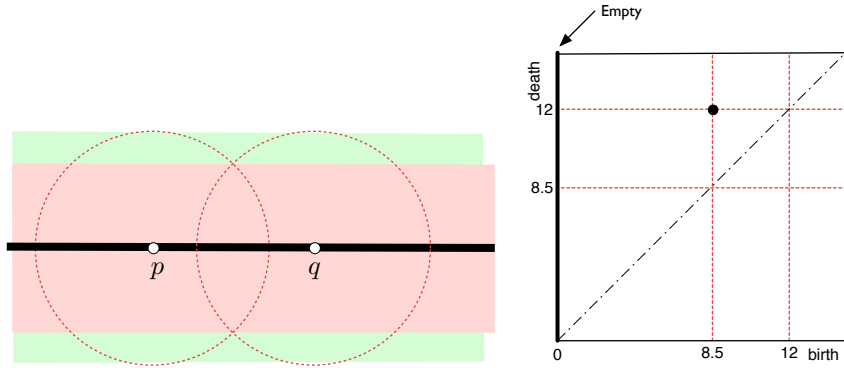


FIGURE 5.10: Kernel persistence diagram of two local equivalent points, given \mathbb{X} . \mathbb{X} is drawn as a solid line. Thickening the space \mathbb{X} is illustrated by shaded regions.

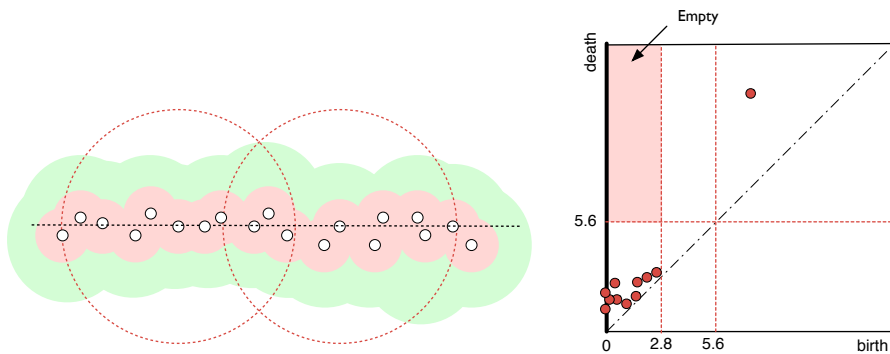


FIGURE 5.11: Kernel persistence diagram of two local equivalent points, given U . Thickening the sample U is illustrated by shaded regions.

Examples. Here we give two examples applying the topological inference theorem.

For the first example, suppose we know the space \mathbb{X} as shown in Figure 5.10, where $\rho = 8.5$. We compute the (co)kernel persistence diagrams by thickening the space \mathbb{X} . Then the given two points have the same local structure iff the y -axes of their (co)kernel persistence diagrams contain no points above 0. In reality, we are only given sample U instead of \mathbb{X} , we then compute the (co)kernel persistence diagrams by thickening U , where $\epsilon = 2.8 < \rho/3$. Then two points have the same local structure iff the rectangles $(\epsilon, 2\epsilon)$ in the diagrams are empty. This is shown in Figure 5.11.

For the second example, suppose \mathbb{X} is given where $\rho = 7$, shown in Figure 5.12. Then two points have different local structure iff the y -axes contain points of the diagrams above

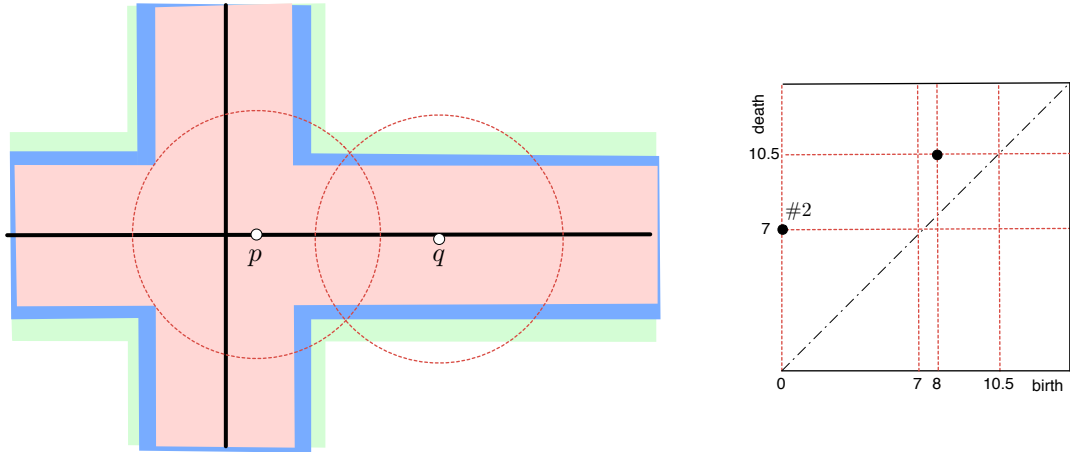


FIGURE 5.12: Kernel persistence diagram of two points that are not local equivalent, given \mathbb{X} . \mathbb{X} is drawn as a solid cross. Thickening the space \mathbb{X} is illustrated by shaded regions. A number, i.e., #2, labeling a point in the persistence diagram indicates its multiplicity.

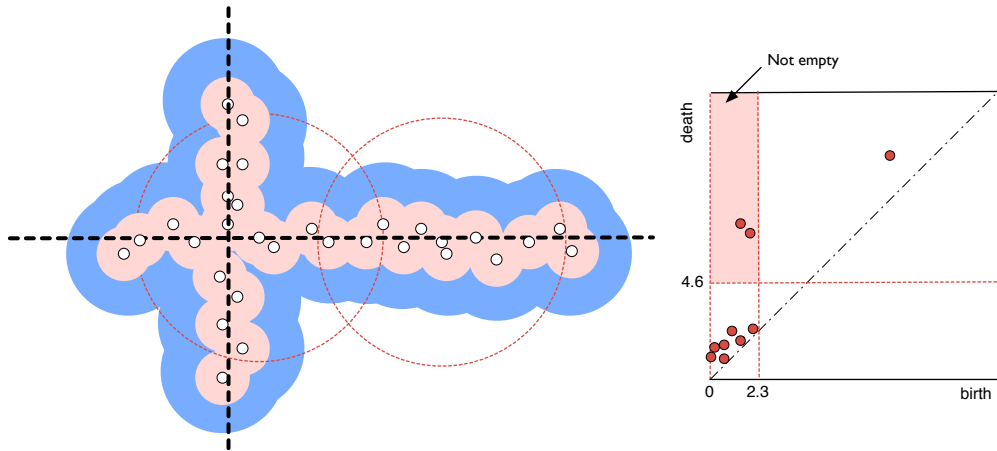


FIGURE 5.13: Kernel persistence diagram of two points that are not local equivalent, given U . Thickening the sample U is illustrated by shaded regions.

0. Correspondingly, if \mathbb{X} is unknown but U is given, where $\epsilon = 2.3 < \rho/3$, then two points have different local structure iff the rectangles $(\epsilon, 2\epsilon)$ in the diagrams are not empty. This is shown in Figure 5.13.

5.4 Geometric Lower Bound

In this section we relate the topological conditions under which points could be assigned to strata to geometric conditions. Most of the effort will involve lower bounding $\rho(p, q, r)$ with geometric features of \mathbb{X} . Specifically, local versions of *reach* and a quantity related to the gradient of $d_{\mathbb{X}}$.

For the geometric condition to make sense we will need to assume that \mathbb{X} is in fact a smooth manifold with boundary. We replace \mathbb{X} with \mathbb{X}_δ for some vanishingly small thickening parameter δ , and smooth the corners, $\mathbb{X} \equiv \mathbb{X}_\delta$.

5.4.1 Absolute Homology Modules

Before providing a geometric lower bound for $\rho(p, q, r)$, we must first prove a technical lemma using some simple algebraic topology. Recall that $\sigma(p, r)$ is the feature size of the relative homology persistence module $\{H(B_p^{\mathbb{X}}, \partial B_p^{\mathbb{X}})\}$. On the other hand, the same thickening process also defines two absolute homology persistence modules, $\{H(B_p^{\mathbb{X}})\}$ and $\{H(\partial B_p^{\mathbb{X}})\}$. We let $\sigma_i(p, r)$ and $\sigma_b(p, r)$ denote the feature sizes of these modules. Similarly, we define $\sigma_i(p, q, r)$ and $\sigma_b(p, q, r)$, respectively, to be the feature sizes of the absolute homology persistence modules $\{H(B_{pq}^{\mathbb{X}})\}$ and $\{H(\partial B_{pq}^{\mathbb{X}})\}$.

Theorem 5.4.1 (Relative/Absolute Lemma). *The feature size of each relative module is at least the minimum of those of its two associated absolute modules:*

$$\begin{aligned}\sigma(p, r) &\geq \min\{\sigma_i(p, r), \sigma_b(p, r)\}, \\ \sigma(p, q, r) &\geq \min\{\sigma_i(p, q, r), \sigma_b(p, q, r)\}.\end{aligned}$$

Proof. We prove the first equality; the second can then be proven with only minor notational adjustment. For any two non-negative reals $\alpha < \beta$, and for each homological

dimension $i \geq 0$, we then consider the following commutative diagram:

$$\begin{array}{ccccccccc}
\mathbf{H}_i(\partial B_p^{\mathbb{X}}(\alpha)) & \rightarrow & \mathbf{H}_i(B_p^{\mathbb{X}}(\alpha)) & \rightarrow & \mathbf{H}_i(B_p^{\mathbb{X}}(\alpha), \partial B_p^{\mathbb{X}}(\alpha)) & \rightarrow & \mathbf{H}_{i-1}(\partial B_p^{\mathbb{X}}(\alpha)) & \rightarrow & \mathbf{H}_{i-1}(B_p^{\mathbb{X}}(\alpha)) \\
\downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\
\mathbf{H}_i(\partial B_p^{\mathbb{X}}(\beta)) & \rightarrow & \mathbf{H}_i(B_p^{\mathbb{X}}(\beta)) & \rightarrow & \mathbf{H}_i(B_p^{\mathbb{X}}(\beta), \partial B_p^{\mathbb{X}}(\beta)) & \rightarrow & \mathbf{H}_{i-1}(\partial B_p^{\mathbb{X}}(\beta)) & \rightarrow & \mathbf{H}_{i-1}(B_p^{\mathbb{X}}(\beta))
\end{array} \tag{5.3}$$

where the vertical maps are induced by the inclusion $X_\alpha \hookrightarrow X_\beta$ and the two rows come from the long exact sequences of the pairs $(B_p^{\mathbb{X}}(\alpha), \partial B_p^{\mathbb{X}}(\alpha))$ and $(B_p^{\mathbb{X}}(\beta), \partial B_p^{\mathbb{X}}(\beta))$ ([82]).

Suppose that the middle vertical map fails to be an isomorphism. Then the Five-Lemma ([82], p.140) tells us that at least one of the other four vertical maps will also fail to be an isomorphism. In other words, any change within the relative module must be accompanied by a simultaneous change in at least one of the two absolute modules. The inequality follows. \square

This theorem together with the definition of $\rho(p, q, r)$ implies the following inequality

$$\rho(p, q, r) \geq \min\{\sigma_i(p, r), \sigma_b(p, r), \sigma_i(p, q, r), \sigma_b(p, q, r)\}. \tag{5.4}$$

5.4.2 Geometric Lower Bounds

We now provide geometric lower bounds for $\sigma_i(p, r), \sigma_b(p, r), \sigma_i(p, q, r), \sigma_b(p, q, r)$ which results in the main theorem in this section.

We first define a few quantities. Recall that the *medial axis* \mathcal{M} of an embedded space \mathbb{X} is the subset of the ambient space consisting of all points which have at least two nearest neighbors on \mathbb{X} , and that the *reach* τ of \mathbb{X} is defined by $\tau = \inf_{x \in \mathbb{X}} \text{dist}(x, \mathcal{M})$. The *local feature size* of a point $a \in \mathbb{X}$ is the distance of a to the medial axis, that is, $\text{lfs}(a) = \text{dist}(a, \mathcal{M})$.

We fix notation for the following four intersections of \mathcal{M} with different subsets of the

ambient space:

$$\mathcal{M}(p, r) = \mathcal{M} \cap B_r(p)$$

$$\mathcal{M}_0(p, r) = \mathcal{M} \cap \partial B_r(p)$$

$$\mathcal{M}(p, q, r) = \mathcal{M} \cap B_r(p) \cap B_r(q)$$

$$\mathcal{M}_0(p, q, r) = \mathcal{M} \cap \partial(B_r(p) \cap B_r(q))$$

We then define a notion of reach for each of the above quantities:

$$\tau(p, r) = \inf_{x \in \mathbb{X}} \text{dist}(x, \mathcal{M}(p, r))$$

$$\tau_0(p, r) = \inf_{x \in \mathbb{X}} \text{dist}(x, \mathcal{M}_0(p, r))$$

$$\tau(p, q, r) = \inf_{x \in \mathbb{X}} \text{dist}(x, \mathcal{M}(p, q, r))$$

$$\tau_0(p, q, r) = \inf_{x \in \mathbb{X}} \text{dist}(x, \mathcal{M}_0(p, q, r)).$$

Note that all four of these quantities are upper bounds on τ itself.

Letting $\nabla d_{\mathbb{X}}$ be shorthand for the gradient of $d_{\mathbb{X}}$, we define the following subset of $\partial B_r(p)$:

$$G(p, r) = \{y \in \partial B_r(p) \mid \nabla d_{\mathbb{X}}(y) \perp \partial B_r(p)\},$$

and then set $\eta(p, r) = \inf_{x \in \mathbb{X}} \text{dist}(x, G(p, r))$. We similarly define $G(p, q, r)$ and $\eta(p, q, r)$,

$$G(p, q, r) = \{y \in \partial(B_r(p) \cap B_r(q)) \mid \nabla d_{\mathbb{X}}(y) \perp \partial(B_r(p) \cap B_r(q))\},$$

$$\eta(p, q, r) = \inf_{x \in \mathbb{X}} \text{dist}(x, G(p, q, r)).$$

Since we assume that \mathbb{X} is a smooth manifold with boundary, the gradient at the corners of the intersections of two balls are well-defined.

The following Lemma will be used to lower bound ρ .

Lemma 5.4.2 (Deformation Lemmas). *The following four claims all hold for every small enough $\delta > 0$. In each of the claims, the homotopy equivalence is given by a deformation*

retraction:

$$\forall \alpha < \min\{\tau(p, r), \eta(p, r)\}, (\mathbb{X}_\alpha \cap B_r(p)) \simeq (\mathbb{X}_\delta \cap B_r(p)),$$

$$\forall \alpha < \min\{\tau_0(p, q, r), \eta(p, r)\}, (\mathbb{X}_\alpha \cap \partial B_r(p)) \simeq (\mathbb{X}_\delta \cap \partial B_r(p)),$$

$$\forall \alpha < \min\{\tau(p, q, r), \eta(p, q, r)\}, (\mathbb{X}_\alpha \cap B_r(p) \cap B_r(q)) \simeq (\mathbb{X}_\delta \cap B_r(p) \cap B_r(q)),$$

$$\forall \alpha < \min\{\tau_0(p, q, r), \eta(p, q, r)\}, (\mathbb{X}_\alpha \cap \partial(B_r(p) \cap B_r(q))) \simeq (\mathbb{X}_\delta \cap \partial(B_r(p) \cap B_r(q))).$$

Proof. All four claims follow from Stratified Morse Theory [59]. We prove only the first claim; the other three can be proven with only slight modifications. Consider the stratification of $B_r(p)$ with singular set $\Sigma = \mathcal{M}(p, r) \cup \partial B_r(p)$ and whatever further decomposition of Σ is needed. Setting $d = d_{\mathbb{X}}|_{B_r(p)} : B_r(p) \rightarrow \mathbb{R}$, we note that the sets $X_\alpha \cap B_r(p)$ are simply the sublevel sets of d for various parameters α . Generically, d will be a Stratified Morse function on $B_r(p)$ with its above stratification. Consider the set H of all critical points of d which have positive d -value.

We claim $H \subset (\mathcal{M}(p, r) \cup G(p, r))$: to see this, we suppose $y \in H$ and we assume first that y is in the interior of $B_r(p)$. Then y is also a critical point of the globally defined function $d_{\mathbb{X}}$, and since $d(x) = d_{\mathbb{X}}(x) > 0$, we know that $y \in \mathcal{M}$. Since y is also in $B_r(p)$ by assumption, we know in fact that $y \in \mathcal{M}(p, r)$. On the other hand, suppose that $y \in \partial B_r(p)$; we can also assume that $y \notin \mathcal{M}(p, r)$ or we are already done. Then by definition y is a critical point of the restriction of $d_{\mathbb{X}}$ to $\partial B_r(p)$. Since the gradient of this latter function is simply the projection of $\nabla d_{\mathbb{X}}$ onto $\partial B_r(p)$, we can conclude $y \in G(p, r)$.

In other words, if $\alpha < \{\tau(p, r), \eta(p, r)\}$, then $(\mathbb{X}_\alpha \cap B_r(p)) \cap H = \emptyset$, and hence the interval $[\delta, \alpha]$ contains no critical values of d . The claim then follows from the first fundamental theorem of Stratified Morse Theory [59]. \square

We now state the geometric lower bound on $\rho(p, q, r)$.

Theorem 5.4.3 (Geometric lower bound). *If we define*

$$\gamma = \gamma(p, q, r) = \min\{\tau(p, r), \tau(p, q, r), \eta(p, r), \eta(p, q, r)\},$$

then $\rho(p, q, r) \geq \gamma(p, q, r)$.

Proof. Note that $\tau(p, r) \leq \tau_0(p, r)$ and $\tau(p, q, r) \leq \tau_0(p, q, r)$ so we need not consider $\tau_0(p, r)$ and $\tau_0(p, q, r)$.

Recall $\sigma_i(p, r)$ and $\sigma_b(p, r)$ were defined to be the feature sizes of the persistence modules $\{H(B_p^{\mathbb{X}}(\alpha))\}$ and $\{H(\partial B_p^{\mathbb{X}}(\alpha))\}$, respectively.

By the first and second of the Deformation Lemmas the following holds

$$\sigma_i(p, r), \sigma_b(p, r) \geq \min\{\tau(p, r), \eta(p, r)\}.$$

For the same reason

$$\sigma_i(p, q, r), \sigma_b(p, q, r) \geq \min\{\tau(p, q, r), \eta(p, q, r)\}.$$

These inequalities, together with (5.4)

$$\rho(p, q, r) \geq \min\{\sigma_i(p, r), \sigma_b(p, r), \sigma_i(p, q, r), \sigma_b(p, q, r)\},$$

proves the theorem, $\rho(p, q, r) \geq \gamma(p, q, r)$. □

In Appendix B we provide more geometric intuitions relating the resolution r of the balls with the reach τ of the topological space. By putting constraints on the resolution parameter, that is, suppose $r < \tau$, we are able to show that our geometric term, γ , is only related to $\tau(p, r)$ and $\tau(p, q, r)$.

5.5 Probabilistic Inference Theorem

The topological inference of Section 5.3 states conditions under which the point sample U can be used to infer stratification properties of the space \mathbb{X} . The basic condition is the

Hausdorff distance between the two is small. In this section, we state two probabilistic models for generating the point sample U and provide an estimate of how large this point sample should be, to infer stratification properties of the space \mathbb{X} with a quantified measure of confidence.

We will assume \mathbb{X} to be compact. The stratified space \mathbb{X} can contain singularities and varying dimensions. This requires some care in the sampling design. Consider a sheet of area one, punctured by a line of length one, sampling from a naively constructed uniform measure on this space would result in no points being sampled from the line. This same issue arose and was dealt with in [89], although in a slightly different approach than we will develop.

The first sampling strategy is to remove the problems of singularities and varying dimension by replacing \mathbb{X} by a slightly thickened version $\mathbb{X} \equiv \mathbb{X}_\delta$. We assume that \mathbb{X} is embedded in \mathbb{R}^k for some k . This new space is a smooth manifold with boundary and our point sample is a set of n points drawn identically and independently from the uniform measure $\mu(\mathbb{X})$ on \mathbb{X} , $U = \{x_1, \dots, x_n\} \stackrel{i.i.d.}{\sim} \mu(\mathbb{X})$. This model can be thought of as placing an appropriate measure on the highest dimensional strata to ensure that lower dimensional strata will be sampled from. In the example of the sheet punched through with a line, the thickened line and sheet will be three dimensional objects. We call this model M_1 .

The second sampling strategy is to deal with the problem of varying dimensions using a mixture model. In the example of the sheet and line, a uniform measure would be placed on the sheet, while another uniform measure would be placed on the line, and a probability mixture would be placed on the two measures, for example, each measure can be drawn with probability $1/2$. We now formalize this approach. Consider each (non-empty) i -dimensional stratum $\mathbb{S}_i = \mathbb{X}_i - \mathbb{X}_{i-1}$ of \mathbb{X} . All strata that are included in the closure of some higher-dimensional strata are not considered in the model. A uniform measure is assigned to each stratum considered in the model, $\mu_i(\mathbb{S}_i)$, this is possible since each

stratum is compact. We assume a finite number of strata K and assign to each stratum a probability $p_i = 1/K$. This implies the following density

$$f(x) = \sum_{j=1}^K \frac{1}{K} \nu_j(X = x),$$

where ν_i is the density corresponding to measure μ_i . The point sample is generated from the following model: $U = \{x_1, \dots, x_n\} \stackrel{i.i.d.}{\sim} f(x)$. We call this model M_2 .

The first model replaces a stratified space with its thickened version, which enables us to place a uniform measure on the thickened space. Although this replacement makes it convenient for sampling, it does not sample directly from the actual space. The second model samples from the actual space, however the sample is not uniform on \mathbb{X} with respect to Lebesgue measure.

Our first main theorem is the probabilistic analogue of Theorem 5.3.3. An immediate consequence of this theorem is that, for two points $p, q \in U$, we can infer whether $p \sim_r q$ with probability $1 - \xi$. The confidence level $1 - \xi$ will be a function of the size of the point sample.

Theorem 5.5.1 (Local Probabilistic Sampling Theorem). *Let $U = \{x_1, x_2, \dots, x_n\}$ be drawn from either model M_1 or M_2 . For a fixed pair of points $p, q \in U$ if*

$$n \geq \frac{1}{\alpha_{new}} \left(\log \frac{1}{\alpha_{new}} + \log \frac{1}{\xi} \right),$$

where $\alpha_{new} = \inf_{x \in \mathbb{X}} \frac{\text{vol}(B_{\rho/24}(x) \cap \mathbb{X})}{\text{vol}(\mathbb{X})}$, with probability greater than $1 - \xi$ we can infer whether $p \sim_r q$.

The above is a consequence of the fact that U is an ϵ -approximation with $\epsilon \leq \rho/3$ with probability $1 - \xi$, where $\rho = \min\{\rho(p, q, r), \rho(q, p, r)\}$.

Proof. Let U be a finite collection of points $x_1, x_2, \dots, x_n \in \mathbb{R}^k$. U is ϵ -dense with respect to \mathbb{X} if $\mathbb{X} \subseteq U^\epsilon$ or equivalently U is an ϵ -cover of \mathbb{X} . Let $C(\epsilon)$ be the ϵ -covering number

of \mathbb{X} , the minimum number of sets $B_\epsilon \cap \mathbb{X}$ that cover \mathbb{X} , where B_ϵ stand for balls of radius ϵ . Let $P(\epsilon)$ be the ϵ -packing number of \mathbb{X} , the maximum number of sets $B_\epsilon \cap \mathbb{X}$ that can be packed into \mathbb{X} without overlap.

For any two points $p, q \in \mathbb{X}$ where $\rho = \min\{\rho(p, q, r), \rho(q, p, r)\} > 0$, we consider a cover of \mathbb{X} by balls of radius $\rho/12$. If we have a sample point in each $\rho/12$ -ball intersecting \mathbb{X} , we have a ϵ -approximation such that $\epsilon \leq 4(\rho/12) = \rho/3$. This satisfies the condition of the topological inference theorem, therefore we can infer the local structure between p and q .

The following two results from [88] will be useful in computing the number of points n needed to be sampled to obtain an ϵ -approximation.

Lemma 5.5.2 (Lemma 5.1 in [88]). *Let $\{A_1, A_2, \dots, A_l\}$ be a finite collection of measurable sets with probability measure μ on $\cup_{i=1}^l A_i$, such that for all A_i , $\mu(A_i) > \alpha$. Let $U = \{x_1, x_2, \dots, x_n\}$ be drawn i.i.d. according to μ . Then if $n \geq \frac{1}{\alpha}(\log l + \log \frac{1}{\xi})$, with probability $1 - \xi$, $\forall i, U \cap A_i \neq \emptyset$.*

Lemma 5.5.3 (Lemma 5.2 in [88]). *Let $C(\epsilon)$ be the covering number of an ϵ -cover of \mathbb{X} and $P(\epsilon)$ be the packing number of an ϵ -packing, then*

$$P(2\epsilon) \leq C(2\epsilon) \leq P(\epsilon).$$

We consider a cover of \mathbb{X} by balls of radius $\rho/12$. Let $\{y_i\}_{i=1}^l \in \mathbb{X}$ be the centers of such balls that constitute a minimal cover. Let $A_i = B_{\rho/12}(y_i) \cap \mathbb{X}$. Applying Lemma 5.5.2, we obtain the estimate

$$n \geq \frac{1}{\alpha} \left(\log l + \log \frac{1}{\xi} \right),$$

where l is the $\rho/12$ -covering number, and $\alpha = \min_i \frac{\text{vol}(A_i)}{\text{vol}(\mathbb{X})} = \min_i \frac{\text{vol}(B_{\rho/12}(y_i) \cap \mathbb{X})}{\text{vol}(\mathbb{X})}$. We now focus on bounding parameters l and α .

Let $\alpha_{new} = \inf_{x \in \mathbb{X}} \frac{\text{vol}(B_{\rho/24}(x) \cap \mathbb{X})}{\text{vol}(\mathbb{X})}$. Applying Lemma 5.5.3,

$$l = C(\rho/12) \leq P(\rho/24) \leq \frac{\text{vol}(\mathbb{X})}{\text{vol}(B_{\rho/24} \cap \mathbb{X})} \leq \frac{1}{\alpha_{new}}.$$

On the other hand, since $\alpha \geq \alpha_{new}$, we have

$$\frac{1}{\alpha} \leq \frac{1}{\alpha_{new}}.$$

Bounding l and α with α_{new} proves the result. □

To extend the local sampling theorem which holds for any two points $p, q \in U$, to a global theorem over all pairs of points $p, q \in U$, we change the sampling resolution. Let ρ_{min} be the minimum ρ for all pairs of points $p, q \in \mathbb{X}$ ($\rho_{min} > 0$). We cover the space with $\rho_{min}/12$ -balls and the using the same proof we obtain the following global sampling result.

Theorem 5.5.4 (Global Probabilistic Sampling Theorem). *Let $U = \{x_1, x_2, \dots, x_n\}$ be drawn from either model M_1 or M_2 . For all pairs of points $p, q \in U$ if*

$$n \geq \frac{1}{\alpha_{new}} \left(\log \frac{1}{\alpha_{new}} + \log \frac{1}{\xi} \right),$$

where $\alpha_{new} = \inf_{x \in \mathbb{X}} \frac{\text{vol}(B_{\rho_{min}/24}(x) \cap \mathbb{X})}{\text{vol}(\mathbb{X})}$, with probability greater than $1 - \xi$ we can infer whether $p \sim_r q$, where $\rho_{min} = \min_{p, q \in U} (\min\{\rho(p, q, r), \rho(q, p, r)\})$.

5.6 Algorithm

Given a point cloud sampled from a stratified space we can use Corollary 5.3.4 to cluster points belonging to a common stratum. That is, once we determine local equivalences among all possible pairs of points in U according to Theorem 5.3.3, we can cluster points into their common strata. Determining local equivalence of two nearby points $p, q \in U$

translates into computing the persistence diagrams $\text{Dgm}(\ker \phi^U)$ and $\text{Dgm}(\text{cok } \phi^U)$, of two persistence modules, $\{\ker \phi_\alpha^U\}$ and $\{\text{cok } \phi_\alpha^U\}$.

We first give a clustering strategy based on assigning weights in an adjacency graph in Section 5.6.1. We then describe an algorithm to compute $\text{Dgm}(\ker \phi^U)$ and $\text{Dgm}(\text{cok } \phi^U)$. We substitute the homology map ϕ^U involving the point cloud with the homology map ψ (described in Section 5.6.2) involving simplicial complexes, and compute $\text{Dgm}(\ker \psi)$ and $\text{Dgm}(\text{cok } \psi)$ instead. Finally, we give a theorem that demonstrates the correctness of the above substitution in Section 5.6.3.

5.6.1 Clustering

We give our clustering strategy as follows. Given a point cloud U sampled from \mathbb{X} , fixing radius r , we build a graph where each node in the graph corresponds to a point in U . Two points $p, q \in U$ (where $\|p - q\| \leq 2r$) are connected by an edge if both $\phi^\mathbb{X}(p, q, r)$ and $\phi^\mathbb{X}(q, p, r)$ are isomorphisms, equivalently, if $\text{Dgm}(\ker \phi^U)(\epsilon, 2\epsilon)$ and $\text{Dgm}(\text{cok } \phi^U)(\epsilon, 2\epsilon)$ are empty. The connected components of the resulting graph are our clusters. A more detailed statement of this procedure is given in pseudo-code, see Algorithm 3. Note, the connectivity of the graph is encoded by a weight matrix, and our clustering strategy is based on a 0/1-weight assignment.

5.6.2 Diagram Computation

We now describe the computation of the diagrams $\text{Dgm}(\ker \phi^U)$ and $\text{Dgm}(\text{cok } \phi^U)$. Recall the homology map ϕ_α^U is defined as,

$$\phi_\alpha^U : \text{H}(B_p^U(\alpha), \partial B_p^U(\alpha)) \rightarrow \text{H}(B_{pq}^U(\alpha), \partial B_{pq}^U(\alpha)).$$

We substitute the homology map ϕ^U involving the point cloud with the homology map ψ involving simplicial complexes, and compute $\text{Dgm}(\ker \psi)$ and $\text{Dgm}(\text{cok } \psi)$ instead.

First we define, for each $\alpha \geq 0$, two pairs of simplicial complexes $L_0(\alpha) \subseteq L(\alpha)$ and

Algorithm 3 Strata-Inference(U, r, ϵ)

```
for all  $p, q \in U$  do
  if  $\|p - q\| > 2r$  then
     $W(p, q) = 0$ 
  else
    Compute  $\text{Dgm}(\ker \phi^U(p, q, r))$  and  $\text{Dgm}(\text{cok } \phi^U(p, q, r))$ 
    Compute  $\text{Dgm}(\ker \phi^U(q, p, r))$  and  $\text{Dgm}(\text{cok } \phi^U(q, p, r))$ 
    if  $\text{Dgm}(\ker \phi^U(p, q, r))(\epsilon, 2\epsilon) \cup \text{Dgm}(\text{cok } \phi^U(p, q, r))(\epsilon, 2\epsilon) \neq \emptyset$  then
       $W(p, q) = 0$ 
    else if  $\text{Dgm}(\ker \phi^U(q, p, r))(\epsilon, 2\epsilon) \cup \text{Dgm}(\text{cok } \phi^U(q, p, r))(\epsilon, 2\epsilon) \neq \emptyset$  then
       $W(p, q) = 0$ 
    else
       $W(p, q) = 1$ 
    end if
  end if
end for
Compute connected components based on  $W$ .
```

$K_0(\alpha) \subseteq K(\alpha)$. Then we define a relative homology map,

$$\psi_\alpha : \mathbf{H}(L(\alpha), L_0(\alpha)) \rightarrow \mathbf{H}(K(\alpha), K_0(\alpha)).$$

We show in the next subsection that $\text{Dgm}(\ker \phi^U) = \text{Dgm}(\ker \psi)$ and $\text{Dgm}(\text{cok } \phi^U) = \text{Dgm}(\text{cok } \psi)$.

To compute the diagrams involving ψ , we reduce various boundary matrices; since we follow very closely the (co)kernel persistence algorithm described in [37], we omit any further details here. We adapt similar shorthand notations by replacing \mathbb{X} by U , i.e., $B_p^U(\alpha)$.

Construct Simplicial Complexes

Constructing simplicial complexes in our algorithm involves taking the nerves of several collections of sets which are derived from a variety of Voronoi diagrams of different spaces. Here we briefly describe these concepts.

Nerves. The *nerve* $N(\mathcal{C})$ of a finite collection of sets \mathcal{C} is defined to be the abstract simplicial complex with vertices corresponding to the sets in \mathcal{C} and with simplices corresponding to all non-empty intersections among these sets; that is, $N(\mathcal{C}) = \{S \subseteq \mathcal{C} \mid \bigcap S \neq \emptyset\}$.

Every abstract simplicial complex can be geometrically realized, therefore the concept of homotopy type makes sense. Under certain conditions, for example whenever the sets in \mathcal{C} are all closed and convex subsets of Euclidean space ([50], p.59), the nerve of \mathcal{C} has the same homotopy type, and thus the same homology groups, as the union of sets in \mathcal{C} .

Voronoi diagram. If U is a finite collection of points in \mathbb{R}^k and $u_i \in U$, then the *Voronoi cell* of u_i is defined to be:

$$V_i = V(u_i) = \{x \in \mathbb{R}^k \mid \|x - u_i\| \leq \|x - u_j\|, \forall u_j \in U\}.$$

The set of cells V_i cover the entire space and form the *Voronoi diagram* of \mathbb{R}^k , denoted as $\text{Voi}(U|\mathbb{R}^k)$. Each V_i restricted to a subset $X \subseteq \mathbb{R}^k$, $V_i \cap X$, cover the space X , and form the *restricted Voronoi diagram*, denoted as $\text{Voi}(U|X)$. For a simplex σ with vertices in U , we set $V_\sigma = \bigcap_{u_i \in \sigma} V_i$.

The nerve of the restricted Voronoi diagram $\text{Voi}(U|X)$ is called the *restricted Delaunay triangulation*, denoted as $\text{Del}(U|X)$. It contains the set of simplices σ for which $V_\sigma \cap X \neq \emptyset$.

Power cells. An important task in our algorithm is the computation of the relative homology groups $H(B_p^U(\alpha), \partial B_p^U(\alpha))$ and $H(B_{pq}^U(\alpha), \partial B_{pq}^U(\alpha))$. Now to compute $H(U_\alpha)$, the absolute homology of the thickened point cloud, we would need only to compute the nerve of the collection of sets $V_i \cap U_\alpha$. This is because each such set is convex and their union obviously equals the space U_α . Such a direct construction will not work in our context, for the simple reason that the sets $V_i \cap \partial B_p^U(\alpha)$ and $V_i \cap \partial B_{pq}^U(\alpha)$ need not be convex.

To get around this problem, we first define $P(\alpha)$, the *power cell* with respect to $B_r(p)$, to be:

$$P(\alpha) = \{x \in \mathbb{R}^k \mid \|x - p\|^2 - r^2 \leq \|x - u\|^2 - \alpha^2, \forall u \in U\}, \quad (5.5)$$

and we set $P_0(\alpha) = B_r(p) - \text{int } P(\alpha)$. To define $Q(\alpha)$, the power cell with respect to $B_r(q)$, we replace p with q in (5.5). Finally, we set the *intersection power cell* as

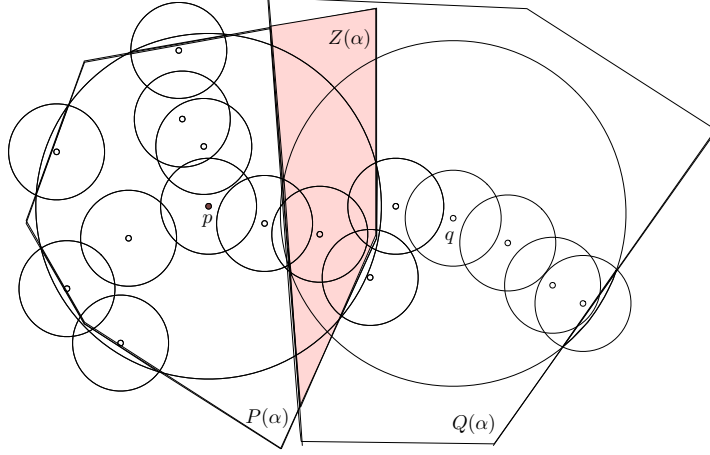


FIGURE 5.14: Illustration of intersection power cell $Z(\alpha)$, as the shaded region. The unshaded convex regions are $P(\alpha)$ and $Q(\alpha)$ respectively.

$Z(\alpha) = P(\alpha) \cap Q(\alpha)$, and $Z_0(\alpha) = (B_r(p) \cap B_r(q)) - \text{int } Z(\alpha)$. This is illustrated in Figure 5.14. We note that $P_0(\alpha)$ and $Z_0(\alpha)$ are both contained in U_α , as can be seen by playing around with the inequalities in their definitions.

Replacing $\partial B_p^U(\alpha)$ with $P_0(\alpha)$ and $\partial B_{pq}^U(\alpha)$ with $Z_0(\alpha)$ has no effect on the relative homology groups in question, as is implied by the following two lemmas. The first lemma was proven in [20]; a proof of the second appears in Appendix D.

Lemma 5.6.1 (Power Cell Lemma). *Assume $B_r(p) - P_0(\alpha) \neq \emptyset$. The identity on $B_p^U(\alpha)$ is a homotopy equivalence of $(B_p^U(\alpha), \partial B_p^U(\alpha))$ and $(B_p^U(\alpha), P_0(\alpha))$ as a map of pairs.*

Lemma 5.6.2 (Intersection Power Cell Lemma). *Assume $B_r(p) \cap B_r(q) - Z_0(\alpha) \neq \emptyset$. The identity on $B_{pq}^U(\alpha)$ is a homotopy equivalence of $(B_{pq}^U(\alpha), \partial B_{pq}^U(\alpha))$ and $(B_{pq}^U(\alpha), Z_0(\alpha))$ as a map of pairs.*

Lune and moon. It can be shown ([20]) that the sets $V_i \cap P_0(\alpha)$ are convex. Unfortunately, it is still possible for some set $V_i \cap Z_0(\alpha)$ to be non-convex. To fix this, we must further divide the Voronoi cells in a manner we now describe.

We consider the hyperplane P of points in \mathbb{R}^k which are equidistant to p and q . This

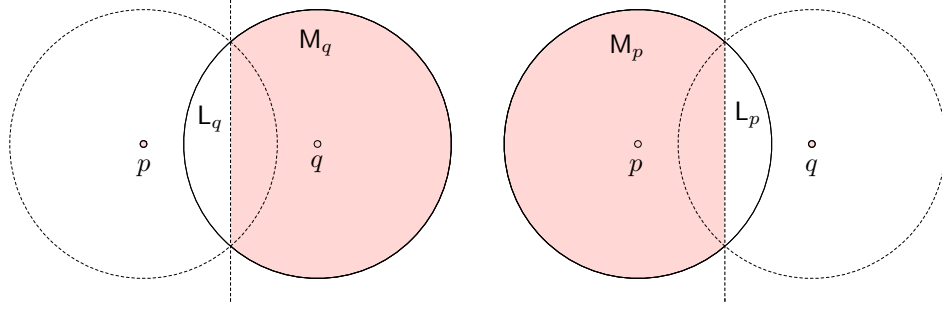


FIGURE 5.15: Illustration of the lune and the moon. The shaded regions are the respective moons. The white regions within solid circles are the respective lunes.

will divide \mathbb{R}^k into two half-spaces; let P_p and P_q denote the half-spaces containing p and q , respectively. We also define the p -lune, $L_p = P_q \cap B_r(p)$, and the p -moon, $M_p = P_p \cap B_r(p)$, as illustrated in Figure 5.15. The lune and the moon divide each Voronoi cell into two parts, defined as the *partial Voronoi cells*, $V_i^L = V_i \cap L_p$ and $V_i^M = V_i \cap M_p$. These sets are convex, assuming they are non-empty, since they are each the intersection of two convex sets. Furthermore, we have the following lemma whose simple but technical proof we omit:

Lemma 5.6.3 (Convexity Lemma). *The sets $V_i^L \cap Z_0(\alpha)$ and $V_i^M \cap Z_0(\alpha)$ are all convex, assuming they are non-empty.*

Of course the nonempty sets among $V_i^L \cap P_0(\alpha)$ and $V_i^M \cap P_0(\alpha)$ will also be convex.

Simplicial complexes. To construct the simplicial complexes needed in our algorithm, first we let \mathcal{A} be the collection of the nonempty sets among $V_i^L \cap B_p^U(\alpha)$ and $V_i^M \cap B_p^U(\alpha)$, and we let \mathcal{A}_0 be the collection of the nonempty sets among $V_i^L \cap P_0(\alpha)$ and $V_i^M \cap P_0(\alpha)$. Note that $\cup \mathcal{A} = B_p^U(\alpha)$ and $\cup \mathcal{A}_0 = P_0(\alpha)$. Taking the nerve of both collections, we define the simplicial complexes $L(\alpha) = N(\mathcal{A})$ and $L_0(\alpha) = N(\mathcal{A}_0)$.

Similarly, we define \mathcal{C} and \mathcal{C}_0 to be the collections of the nonempty sets among, respectively, $V_i^L \cap B_{pq}^U(\alpha)$ and $V_i^M \cap B_{pq}^U(\alpha)$, and $V_i^L \cap Z_0(\alpha)$ and $V_i^M \cap Z_0(\alpha)$. And we define $K(\alpha) = N(\mathcal{C})$ and $K_0(\alpha) = N(\mathcal{C}_0)$.

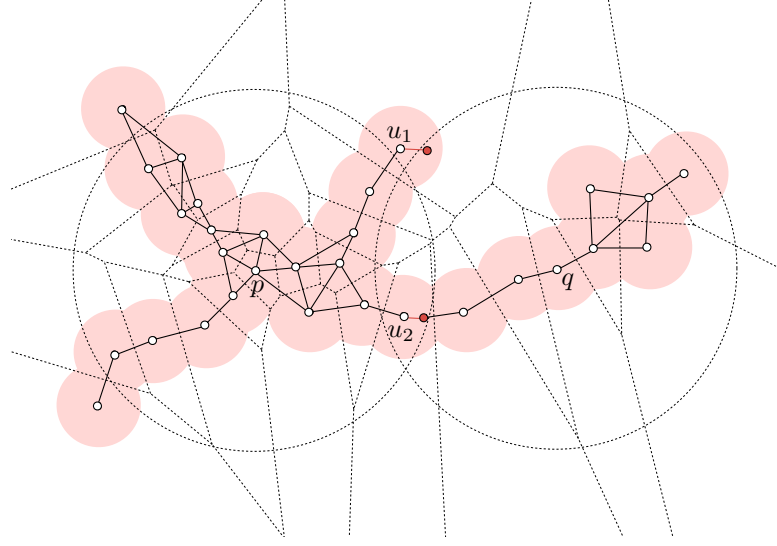


FIGURE 5.16: Illustration of the simplicial complexes constructed around two points p and q . The underlying Voronoi decomposition of the space is shown in thin dotted lines. u_1 and u_2 in U are the points whose restricted Voronoi regions intersect with the lune at non-convex regions.

For simplicity, for a simplex $\sigma \in L(\alpha)$ (similarly for a simplex in L_0 , K and K_0), we define V^σ as the intersection of the partial Voronoi cells that correspond to the vertices of σ . That is, $\sigma \in L(\alpha)$ iff $V^\sigma \cap B_p^U(\alpha) \neq \emptyset$.

An example of the simplicial complexes constructed in \mathbb{R}^2 for a given U are illustrated in Figure 5.16. A direct approach to construct these simplicial complexes runs into difficulties as the corners of the convex sets created by the bisector can be shared by many sets, we defer the technicalities to Appendix C.

Construct ψ

We now construct simplicial analogues ψ_α of the maps ϕ_α^U ,

$$\psi_\alpha : \mathbf{H}(L(\alpha), L_0(\alpha)) \rightarrow \mathbf{H}(K(\alpha), K_0(\alpha)).$$

The containments $L_0(\alpha) \subseteq L(\alpha)$ and $K_0(\alpha) \subseteq K(\alpha)$ are obvious. In order to define ψ_α , we first need the following technical lemma:

Lemma 5.6.4 (Containment Lemma). *Assume that a simplex σ is in $L_0(\alpha)$. If σ is also in $K(\alpha)$, then σ is in $K_0(\alpha)$, as well.*

Proof. . By definition, $\sigma \in L_0(\alpha)$ iff there exists some point $x \in V^\sigma \cap P_0(\alpha)$. We must show that the set $V^\sigma \cap Z_0(\alpha)$ is non-empty. Note that $x \in P_0(\alpha)$ implies that $x \in B_r(p)$, while $x \notin \text{int } P(\alpha)$ implies that $x \notin \text{int } Z(\alpha)$. If $x \in B_r(q)$, then we are done, since $Z_0(\alpha) = B_r(p) \cap B_r(q) - \text{int } Z(\alpha)$.

Otherwise, choose some point $y \in V^\sigma \cap U_\alpha \cap B_r(p) \cap B_r(q)$, which is possible since $\sigma \in K(\alpha)$. Since both x and y belong to the same convex set $V^\sigma \cap U_\alpha \cap B_r(p)$, there exists a directed line segment γ from x to y within this set connecting them. We imagine moving along γ and first we suppose that γ intersects $B_r(q)$ before it intersects $\text{int } Q(\alpha)$. Let z be the first point of intersection. Then $z \in B_r(p) \cap B_r(q)$, $z \notin \text{int } Q(\alpha)$. Therefore $z \in V^\sigma \cap Z_0(\alpha)$. On the other hand, we may prove by contradiction that it is impossible for γ to intersect $Q(\alpha)$ before it intersects $B_r(q)$. Let z' be the first point of such an intersection. Since $z' \in Q(\alpha)$, by definition $\|z' - q\|^2 - r^2 \leq \|z' - u_i\|^2 - \alpha^2, \forall u_i \in U$. Since $z' \in U_\alpha$, $\forall u_i \in \sigma, \|z' - u_i\|^2 - \alpha^2 \leq 0$. Therefore $\|z' - q\|^2 - r^2 \leq \|z' - u_i\|^2 - \alpha^2 \leq 0, \forall u_i \in \sigma$. Since z' is outside $B_r(q)$, $\|z' - q\|^2 - r^2 > 0$. This is a contradiction. \square

To define ψ_α , we first construct a chain map $g = g_\alpha : C(L(\alpha)) \rightarrow C(K(\alpha))$ as follows. Given a simplex $\sigma \in L(\alpha)$, we define $g(\sigma) = \sigma$ if $\sigma \in K(\alpha)$, and $g(\sigma) = 0$ otherwise; we then extend g to a chain map by linearity. Using the Containment Lemma, we see that $g(C(L_0(\alpha))) \subseteq C(K_0(\alpha))$, and thus g descends to a relative chain map $f = f_\alpha : C(L(\alpha), L_0(\alpha)) \rightarrow C(K(\alpha), K_0(\alpha))$. Since f clearly commutes with all boundary operators, it induces a map on relative homology, and this is our $\psi = \psi_\alpha$.

5.6.3 Correctness

We show that our algorithm is correct by proving the following theorem. We describe a proof sketch here and defer all details to Appendix D.

Theorem 5.6.5 (Equivalent Persistence Diagram Theorem). *The persistence diagrams involving simplicial complexes are equal to the persistence diagrams involving the point*

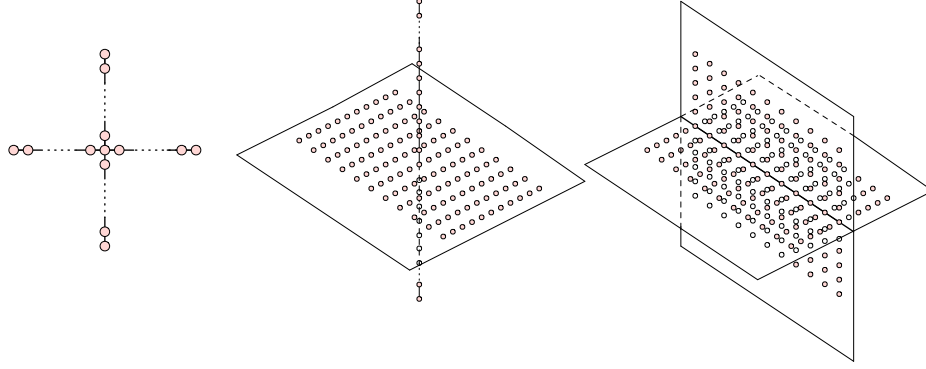


FIGURE 5.17: From left to right: points sampled from a cross; points sampled from a plane intersecting a line; points sampled from two intersecting planes. All points are located on the grid.

cloud, that is, $\text{Dgm}(\ker \phi^U) = \text{Dgm}(\ker \psi)$ and $\text{Dgm}(\text{cok } \phi^U) = \text{Dgm}(\text{cok } \psi)$.

Proof sketch. To prove Theorem 5.6.5, we need to prove the following diagram (as well as similar diagram involving cokernels) commutes and all vertical maps are isomorphisms.

That is, for each pair $\alpha \leq \beta$, we have the following commuting diagram:

$$\begin{array}{ccccccc}
 \dots & \rightarrow & \ker \phi_\alpha^U & \rightarrow & \ker \phi_\beta^U & \rightarrow & \dots \\
 & & \uparrow \cong & & \uparrow \cong & & \\
 \dots & \rightarrow & \ker \psi_\alpha & \rightarrow & \ker \psi_\beta & \rightarrow & \dots
 \end{array} \tag{5.6}$$

Hence the Persistence Equivalence Theorem gives $\text{Dgm}(\ker \phi^U) = \text{Dgm}(\ker \psi)$ and $\text{Dgm}(\text{cok } \phi^U) = \text{Dgm}(\text{cok } \psi)$.

5.7 Simulations

We use a simulation on simple synthetic data with points sampled from grids to illustrate how the algorithm performs. In these simulations we assume we know ϵ , and we run our algorithm for $0 \leq \alpha \leq 2\epsilon$. The data sets are shown in Figure 5.17.

We use the following results to demonstrate that the inference on local structure, at least for these very simple examples, is correct. As shown in Figure 5.18 top, if two

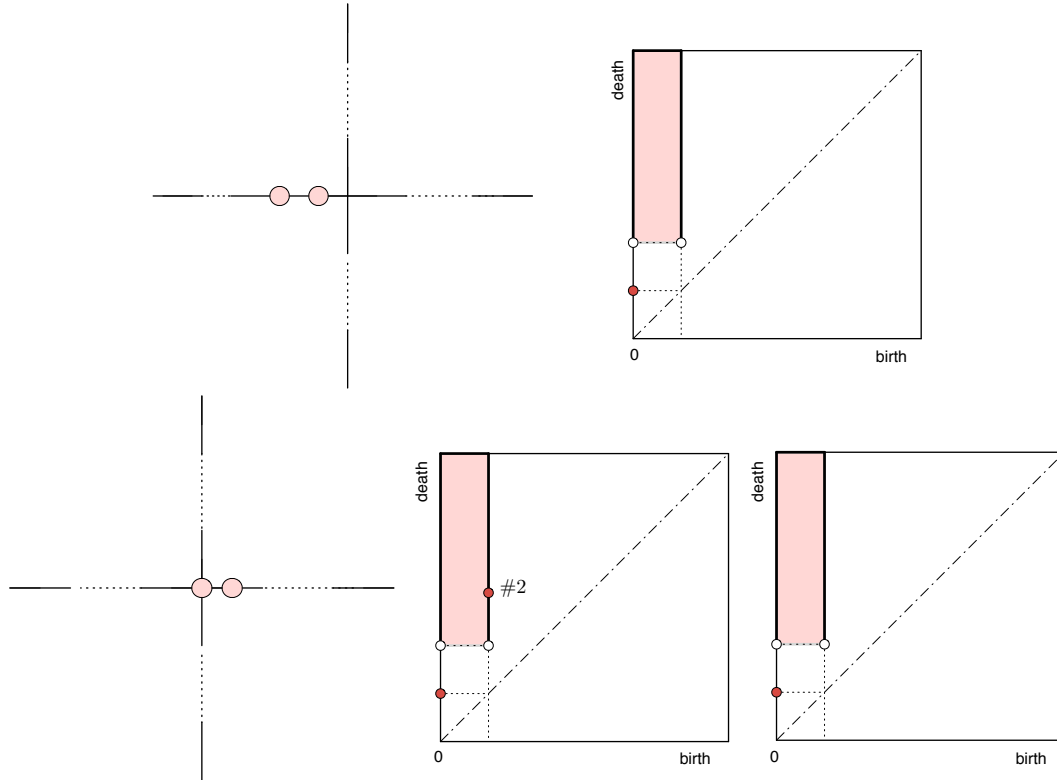


FIGURE 5.18: Top: both points are from 1-strata. Bottom: one point from 0-strata, one point from 1-strata. Left part shows the locations of the points. Right part shows the ker/cok persistence diagram of two points respectively, if the diagrams are the same, only one is shown. A number labeling a point in the persistence diagram indicates its multiplicity.

points are locally equivalent, their corresponding ker/cok persistence diagrams contain the empty quadrant prescribed by our theorems, while in Figure 5.18 bottom, the diagrams associated to two non-equivalent points do not contain such empty quadrants. Similar results are shown for other data sets in Figure 5.19 and Figure 5.20.

5.8 Discussion

As the title of the chapter suggests we have presented a first step towards learning stratified spaces. In the discussion we state some future problems and extensions of interest.

1. **Algorithmic efficiency:** The algorithm to compute the (co)kernel diagrams from the thickened point cloud is based on an adaption of Delaunay triangulation and

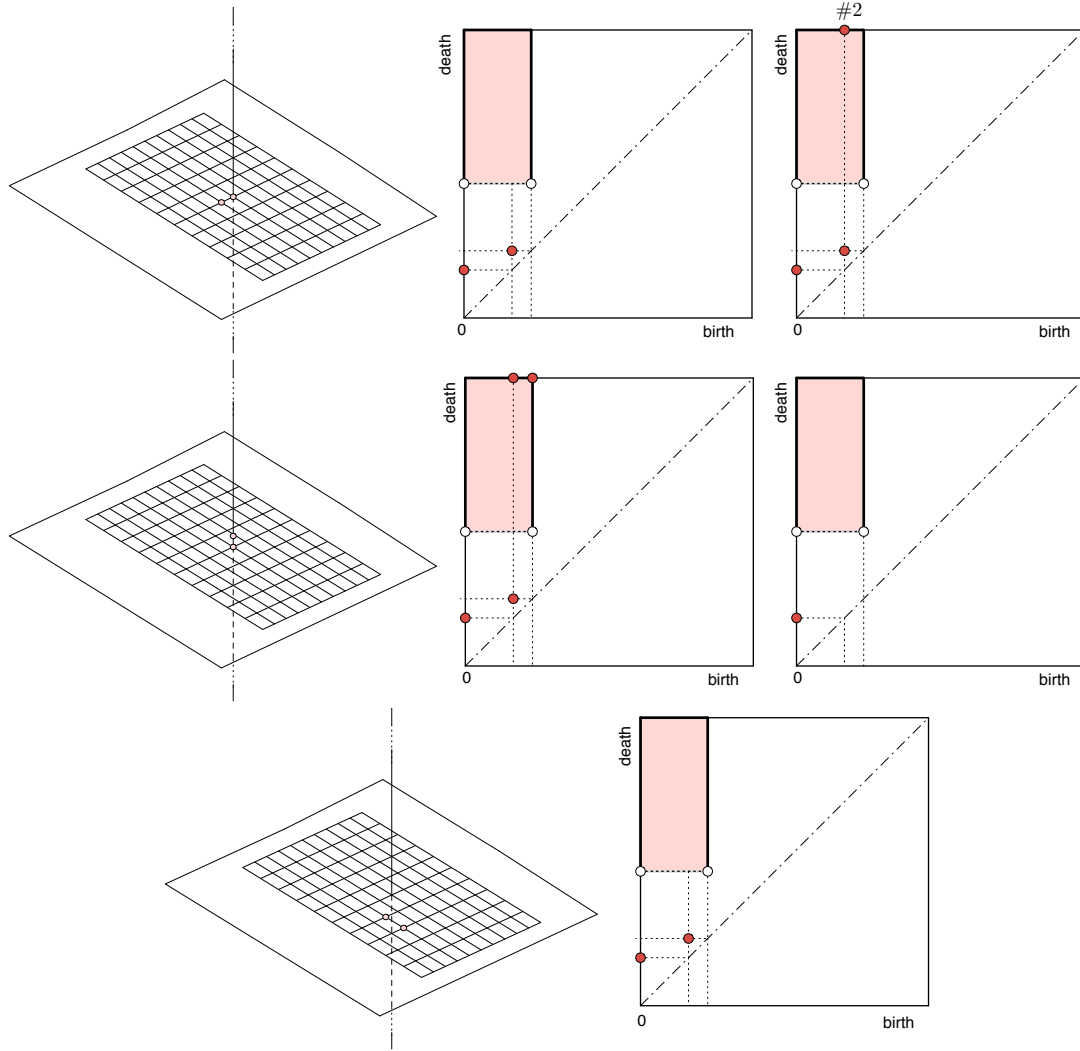


FIGURE 5.19: Top: one point from 0-strata, one point from 2-strata. Middle: one point from 0-strata, one from 1-strata. Bottom: both points are from 2-strata. A number labeling a point in the persistence diagram indicates its multiplicity.

the power-cell construction. This algorithm should be quite slow when the dimensionality of the ambient space is high due to the runtime complexity of Delaunay triangulation. One idea to address this bottleneck is to use Rips or Witness complexes [44]. Another approach is to use dimension reduction techniques such as principal components analysis (PCA) or random projection that approximately preserve distance [31] as a preprocessing step. If the dimension of the ambient space is not very high, we might be able to use faster algorithms to construct Delaunay

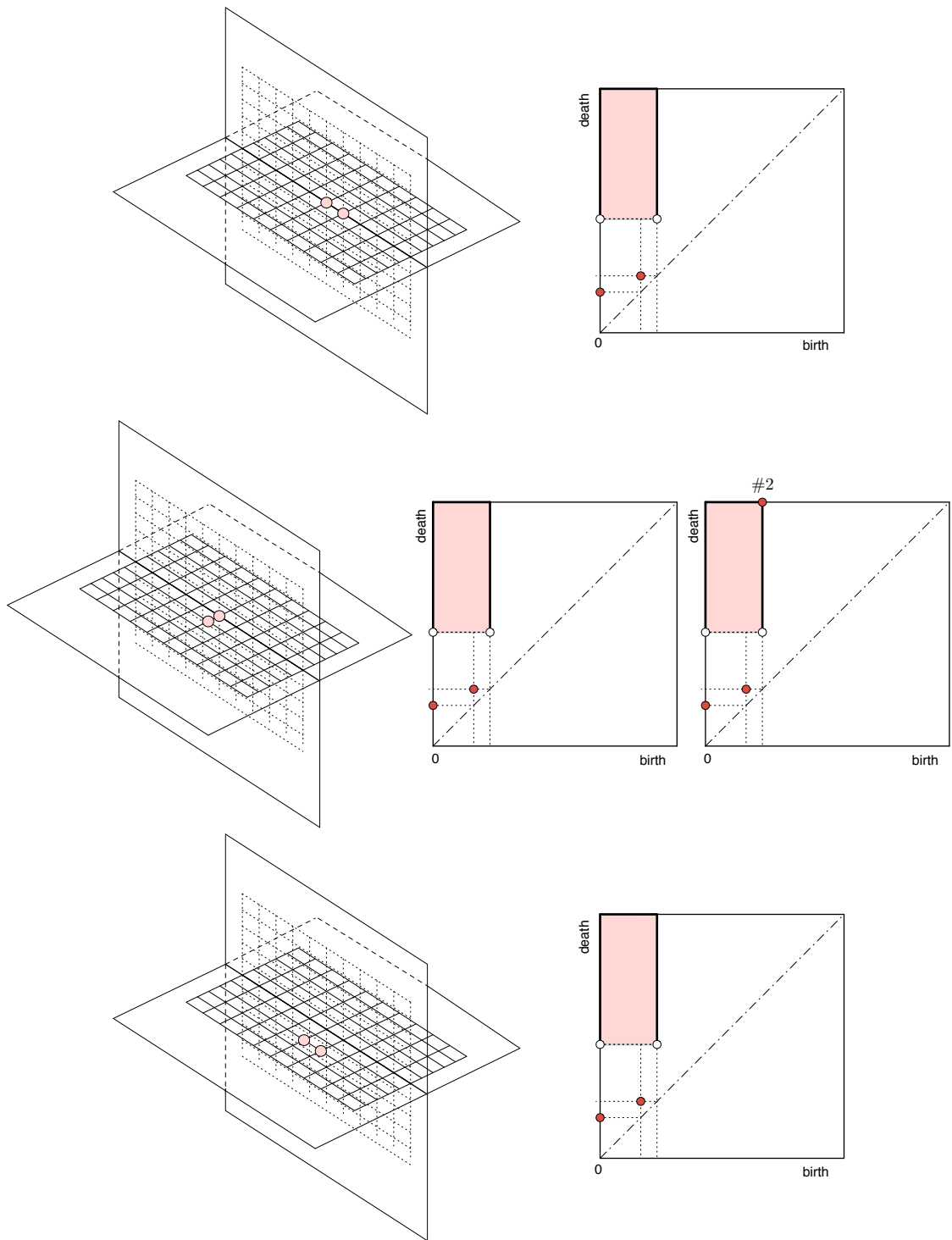


FIGURE 5.20: Top: both points from 1-strata. Middle: one point from 1-strata, one from 2-strata. Bottom: both points are from 2-strata. A number labeling a point in the persistence diagram indicates its multiplicity.

triangulations [21].

2. **Weighting local equivalence:** Currently we use a graph with 0/1 weights based on the local equivalence between two points. If we can suggest a more continuous metric for local equivalence, measuring how similar two points are in terms of their local structure, we can extend this idea to assign fractional weights between points. This would also allow us to use approaches such as Laplacian Eigenmaps [18] to assign points to strata.
3. **Curvature moderated tubes:** Markus J. Pflaum [93] introduced a concept called *curvature moderation* that regulates the behavior of the tangent spaces of a stratum near the boundary. In other words, a stratum is curvature moderate if it curves near the boundary in a controlled way, this includes the higher derivatives of the curvature. This is yet another way to describe how strata and their tubular neighborhood are “glued together nicely”. It would be interesting to connect this concept to our idea of “local reach”.
4. **Noisy data:** Our sampling models draw points from the underlying topological space. A more general model would sample points that are concentrated on the topological space. A version of this type of sampling model is discussed in [88]. It would be of interest to study how well our approach is suited to such a model.
5. **Adaptive sampling conditions:** Throughout this chapter we use ϵ -approximation to characterize the similarity of the point sample to the topological space. There are other approximation criteria that may be interesting to study and may provide better sampling estimates. One such criteria is the adaptive ϵ -sample [45], which is proportional to the local feature size. Another criteria of possible interest is the weak feature size [28].

Appendix to Chapter 5

A Defining the Map ϕ

We give a more precise description of the map

$$\phi = \phi_\alpha^U : H(B_p^U(\alpha), \partial B_p^U(\alpha)) \rightarrow H(B_{pq}^U(\alpha), \partial B_{pq}^U(\alpha)).$$

The definition will be made on the chain level and will be given in terms of singular chains. The map $\phi_\alpha^X : H(B_p^X(\alpha), \partial B_p^X(\alpha)) \rightarrow H(B_{pq}^X(\alpha), \partial B_{pq}^X(\alpha))$ can be defined in a similar fashion.

In this and subsequent sections, we use capital letters to denote topological spaces, i.e. X, Y, U and V .

A.1 Background

We give here some necessary background as well as some material from algebraic topology and homological algebra which will be needed in Appendix D. Most of the descriptions are adapted from [63] and [82].

Chain homotopies. For our purposes, a *chain complex* \mathcal{C} is a sequence of $\mathbb{Z}/2\mathbb{Z}$ -vector spaces C_p , one for each integer p , connected by boundary homomorphisms $\partial_p^C : C_p \rightarrow C_{p-1}$ such that $\partial_{p-1} \circ \partial_p = 0$ for all p . The p -th homology group of such a chain complex is defined by $H_p = \ker \partial_p / \text{im } \partial_{p+1}$.

A *chain map* $\eta : \mathcal{C} \rightarrow \mathcal{D}$ between two chain complexes is a sequence of homomorphisms $\eta_p : C_p \rightarrow D_p$ which commute with the boundary homomorphisms: $\partial_p^D \circ \eta_p = \eta_{p-1} \circ \partial_p^C$. Every chain map induces a map η_* between the homology groups of the two complexes.

A *chain homotopy* F between two chain maps $\eta, \eta' : \mathcal{C} \rightarrow \mathcal{D}$ is a sequence of homomorphisms $F_p : C_p \rightarrow D_{p+1}$ which satisfy the following formula for each p : $\eta - \eta' = \partial_{p+1}^D \circ F_p - F_{p-1} \circ \partial_p^C$. We say that η and η' are *chain homotopic* and note that they must

then induce the same maps on homology: $\eta_* = \eta'_*$. Finally, η is called a *chain homotopy equivalence* if there exists a chain map $\rho : \mathcal{D} \rightarrow \mathcal{C}$ such that $\eta \circ \rho$ and $\rho \circ \eta$ are both chain homotopic to the identity. In this case η and ρ will both induce homology isomorphisms.

Singular homology. The *standard p -simplex* is the subset of \mathbb{R}^{p+1} , given by,

$$\Delta_p = \{(t_0, \dots, t_p) \in \mathbb{R}^{p+1} \mid \sum_{i=0}^p t_i = 1, \forall i, t_i \geq 0\}.$$

The $p + 1$ vertices of Δ_p are points $\{e_i\} \subset \mathbb{R}^{p+1}$ where

$$e_0 = (1, 0, 0, \dots, 0),$$

$$e_1 = (0, 1, 0, \dots, 0),$$

...

$$e_p = (0, 0, 0, \dots, 1).$$

A *singular p -simplex* of a topological space X is a continuous map

$$\delta : \Delta_p \rightarrow X,$$

where Δ_p is the *standard p -simplex*. By taking formal sums of singular simplices (using binary coefficients for our purposes) one forms $C_p(X)$, the *singular chain group* of X in dimension p . Given points a_0, \dots, a_p in some Euclidean space, which need not be independent, there is a unique affine map l of Δ_p that maps the vertices e_i of Δ_p to a_i . This map defines *linear singular simplex* determined by a_0, \dots, a_p , denoted as $l(a_0, \dots, a_p)$. One then defines a boundary homomorphism $\partial_p : C_p(X) \rightarrow C_{p-1}(X)$ by:

$$\partial_p(\delta) = \sum_{i=0}^p (\delta \circ l(\epsilon_0, \dots, \hat{\epsilon}_i, \dots, \epsilon_p)),$$

and defines the singular homology groups $H_p(X)$ as above. A continuous map f from X to another topological space Y induces a chain map $f_{\#} : C_p(X) \rightarrow C_p(Y)$ given by the formula $f_{\#}(\delta) = f \circ \delta$, and a homology map $f_* : H_p(X) \rightarrow H_p(Y)$.

The *minimal carrier* of a singular simplex δ is its image $\delta(\Delta_p)$, and the *minimal carrier* of a singular p -chain $\sum \delta_i$ is the union of the minimal carriers of the δ_i .

Isomorphism between simplicial and singular homology. The (simplicial) homology groups of a simplicial complex K and the singular homology groups of its realization $|K|$ are isomorphic. To show an explicit isomorphism ([82]), we first define a chain map

$$\eta : C(K) \rightarrow C(|K|)$$

as follows [82]: choose a partial ordering of the vertices of K that induces a linear ordering on the vertices of each simplex of K . Orient the simplices of K by using this ordering, and define

$$\eta([v_0, \dots, v_p]) = l(v_0, \dots, v_p),$$

where $v_0 < \dots < v_p$ in the given ordering. We refer to the linear singular simplex $l(v_0, \dots, v_p)$ as a *simplicial linear singular simplex* and it is important in the subsequent sections. The chain map η is in fact a chain equivalence as it has a *chain-homotopy inverse*, for which a specific formula can be found in [52]. Hence the induced homology map η_* provides an isomorphism of simplicial with singular homology.

Barycentric subdivision. Let Sd denotes a barycentric subdivision homomorphism, $\text{Sd} : C_p(X) \rightarrow C_p(X)$. Iterate the map m times, we denote the m -th barycentric subdivision as Sd^m . For technical details, see [63].

A.2 Intersection Map Details

We now give the full and formal definition of the homology map $\phi = \phi_\alpha^U$, starting on the chain level. For compactness, we will use the following shorthand:

$$X = B_p^U(\alpha) = U_\alpha \cap B_r(p),$$

$$B = \partial B_p^U(\alpha) = U_\alpha \cap \partial B_r(p),$$

$$S = B_{pq}^U(\alpha) = U_\alpha \cap B_r(p) \cap B_r(q),$$

$$A = U_\alpha \cap B_r(p) - \text{int}(S),$$

$$U = U_\alpha \cap B_r(p) - S.$$

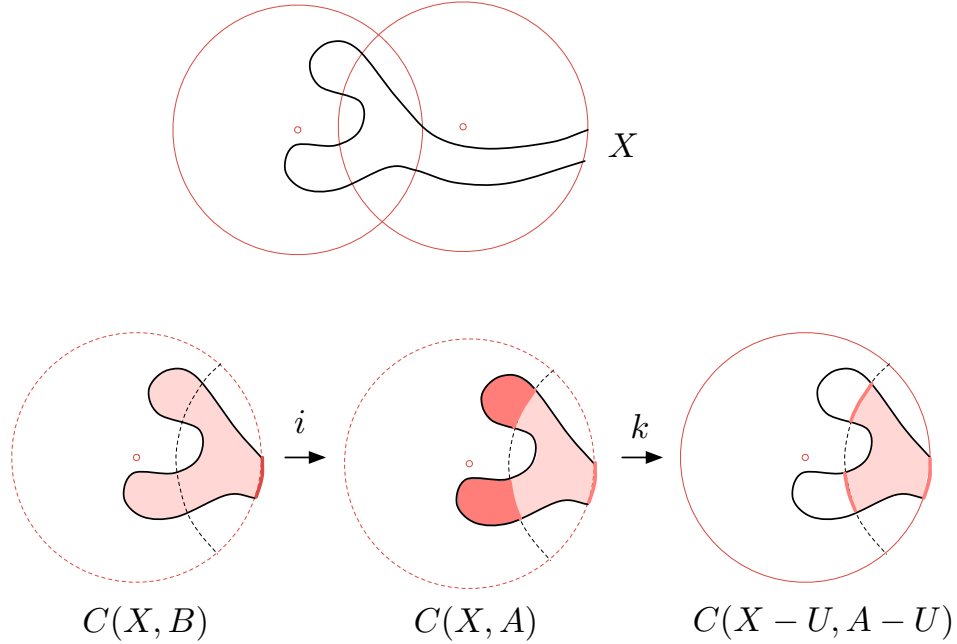


FIGURE 5.21: Definition of j , an example, spaces X , $X - U$ are the light-shaded regions, B and A are the dark-shaded regions.

Note that $X - U = S = B_{pq}^U(\alpha)$ and $A - U = \partial S = \partial B_{pq}^U(\alpha)$. So to define ϕ we need only define a chain map $j : C(X, B) \rightarrow C(X - U, A - U)$ and then take ϕ as the map induced on homology. The map j is defined as the composition $j = k \circ i$. This is illustrated in Figure 5.21. The chain map $i : C(X, B) \rightarrow C(X, A)$ is induced by inclusion on the second factor, while the chain map $k : C(X, A) \rightarrow C(X - U, A - U)$ is an excision, although this latter statement requires further elaboration.

Excisions. The inclusion map of pairs $(X - U, A - U) \rightarrow (X, A)$ is called an **excision** if it induces a homology isomorphism; in this case one says that U can be excised. We will make use of the following two results about excision (see, e.g., [60]):

Theorem A.1. (*Excision Theorem*) *If the closure of U is contained in the interior of A , then U can be excised.*

Theorem A.2. (*Excision Extension*) *Suppose $V \subset U \subset A$ and*

(i) V can be excised.

(ii) $(X - U, A - U)$ is a *deformation retract* of $(X - V, A - V)$.

Then U can be excised.

In our context, the closure of U need not be contained in the interior of A , and so we must define a suitable $V \subset U$. Although there are many ways to do this, one direct way is to choose some small enough positive δ .

$$I = U_\alpha \cap \partial(B_r(p) \cap B_r(q)) \cap \text{cl}(U),$$

$$I_\delta = \{x \in \text{cl}(U) \mid d_I(x) \leq \delta\},$$

$$V = U - I_\delta,$$

where $d_I(x)$ is the distance from x to the set I . It is straightforward to verify that $V \subset U \subset A$ satisfies the hypotheses of Theorem A.2. In other words, the inclusion of pairs $(X - V, A - V) \rightarrow (X, A)$ is an excision; its induced chain map has a chain-homotopy inverse, which we denote as $s : C(X, A) \rightarrow C(X - V, A - V)$. Finally, we define $k = r_\# \circ s$, where $r_\#$ is the chain map induced by the retraction $r : (X - V, A - V) \rightarrow (X - U, A - U)$.

Subdivision. In order to fully carry out the analysis in Appendix D, we must first decompose the maps i and k through subdivision. Given a topological space X and a collection \mathcal{A} of subsets of X whose interiors form an open cover of X , a singular simplex of X is said to be *\mathcal{A} -small* if its image set is entirely contained in a single element of \mathcal{A} [82]. For each dimension p , the chain group $C_p^{\mathcal{A}}(X)$ is the subgroup of $C_p(X)$ spanned by the \mathcal{A} -small singular p -simplices. These groups form a chain complex, with homology $H^{\mathcal{A}}(X)$. Of course, any singular simplex on X can be subdivided into a sum of \mathcal{A} -small simplices, so it is plausible, and in fact true ([63]), that the inclusion $C^{\mathcal{A}}(X) \rightarrow C(X)$ is a chain homotopy equivalence.

Returning to our context, we set $\mathcal{A} = \{X - V, A\}$ and denote by l the chain inclusion $C^{\mathcal{A}}(X, A) \rightarrow C(X, A)$. We also let $\rho : C(X, B) \rightarrow C^{\mathcal{A}}(X, B)$ be the chain homotopy inverse of the chain inclusion $C^{\mathcal{A}}(X, B) \rightarrow C(X, B)$, and let $t : C^{\mathcal{A}}(X, B) \rightarrow C^{\mathcal{A}}(X, A)$ be the chain map induced by inclusion on the second factor. Finally we note that $i = l \circ t \circ \rho$.

We also decompose k as $k = r_{\#} \circ \eta \circ \rho$, where η is the chain homotopy inverse of the chain map $C(X - V, A - V) \rightarrow C(X, A) \rightarrow C^{\mathcal{A}}(X, A)$.

Summary. To summarize, our map $\phi = j_*$, where j is the chain map defined by the following sequence of chain maps

$$j = k \circ i = (r_{\#} \circ \eta \circ \rho) \circ (l \circ t \circ \rho).$$

Since $\rho \circ l$ is homotopic to the identity map, j can be simplified such that $j = r_{\#} \circ \eta \circ t \circ \rho$, that is,

$$C(X, B) \xrightarrow{\rho} C^{\mathcal{A}}(X, B) \xrightarrow{t} C^{\mathcal{A}}(X, A) \xrightarrow{\eta} C(X - V, A - V) \xrightarrow{r_{\#}} C(X - U, A - U).$$

Following the same framework as above, we also define a chain map j' and its induced homology map $\phi' = j'_* : H(B_p^U(\alpha), P_0(\alpha)) \rightarrow H(B_{pq}^U(\alpha), Z_0(\alpha))$ simply by adopting the notation:

$$\begin{aligned} X &= B_p^U(\alpha) = U_{\alpha} \cap B_r(p), \\ B' &= P_0(\alpha), \\ S &= B_{pq}^U(\alpha) = U_{\alpha} \cap B_r(p) \cap B_r(q), \\ A' &= U_{\alpha} \cap B_r(p) - S + Z_0(\alpha), \\ U &= U_{\alpha} \cap B_r(p) - S, \\ I &= U_{\alpha} \cap \partial(B_r(p) \cap B_r(q)) \cap \text{cl}(U), \\ I_{\delta} &= \{x \in \text{cl}(U) \mid d_I(x) \leq \delta\}, \\ V &= U - I_{\delta}, \end{aligned}$$

defining our open cover to be $\mathcal{A}' = \{X - V, A'\}$, and otherwise proceeding exactly as before.

Similarly, we create a chain map f' which induces $\psi' = f'_* : H(|\text{Sd } L|, |\text{Sd } L_0|) \rightarrow H(|\text{Sd } K|, |\text{Sd } K_0|)$, using the notation

$$\begin{aligned} X'' &= |\text{Sd } L|, \\ B'' &= |\text{Sd } L_0|, \\ A'' &= (|\text{Sd } L| - |\text{Sd } K|) \cup |\text{Sd } K_0|, \\ U'' &= |\text{Sd } L| - \text{int } |\text{Sd } K|, \\ I' &= |\text{Sd } K| \cap \text{cl}(U''), \\ I'_\delta &= \{x \in \text{cl}(U'') \mid d_{I''}(x) \leq \delta\}, \\ V'' &= U'' - I'_\delta, \end{aligned}$$

with $\mathcal{A}'' = \{A'', X'' - V''\}$.

B Control of η Parameters

In this section, we show that our geometric term, γ , is only related to $\tau(p, r)$ and $\tau(p, q, r)$, if we put constraints on the resolution parameter r with respect to the reach τ of \mathbb{X} . Again, we assume that \mathbb{X} is a smooth manifold with boundary.

Theorem B.1 states that if the resolution of the balls is finer than the reach of \mathbb{X} , then $\eta(p, r) = \eta(q, r) = \eta(p, q, r) = r$. This implies that

$$\gamma = \gamma(p, q, r) = \min\{\tau(p, r), \tau(p, q, r), \eta(p, r), \eta(p, q, r)\} = \min\{\tau(p, r), \tau(p, q, r), r\}.$$

In other words, if the balls are small enough, γ is only related to several local notions of reach.

Theorem B.1 (Orthogonality Theorem). $\forall r < \tau$, and $\forall p, q \in \mathbb{X}$, we have

$$\eta(p, r) = \eta(q, r) = \eta(p, q, r) = r.$$

To prove this theorem, we first prove the following proposition.

Proposition B.2. *Suppose $r < \tau$ and $p \in \mathbb{X}$, a is any point in $G(p, r)$. Then p is the closest point on \mathbb{X} from a .*

Proof. We prove the proposition by contradiction. If $r < \tau$, we assume p is not the closest point on \mathbb{X} from a . That is, there exists a point $a' \in \mathbb{X}$, $a' \neq p$, such that $\|a - a'\| < \|a - p\| = r$.

We first prove that if such an a' exists, it must lie on the line connecting a and p .

Let the gradient of $d_{\mathbb{X}}$ at a be $\nabla_{d_{\mathbb{X}}}(a)$. Let the tangent plane to $B_r(p)$ at a be T_a . Since $a \in \partial B_r(p) \cap G(p, q, r)$, $\nabla_{d_{\mathbb{X}}}(a)$ is orthogonal to T_a , that is $\nabla_{d_{\mathbb{X}}}(a) \perp T_a$. Let the normal vector at a be N_a , we have $N_a \perp T_a$. Therefore $\nabla_{d_{\mathbb{X}}}(a)$ and N_a are colinear. Since p is the center of $B_r(p)$, N_a is colinear with the line connecting a and p . Therefore $\nabla_{d_{\mathbb{X}}}(a)$ is colinear with the line connecting a and p . If a' is the closet point on \mathbb{X} from a , then $\nabla_{d_{\mathbb{X}}}(a)$ is colinear with the line connecting a' and a . Therefore a' is colinear with a and p .

If such a' exists, there can be two cases, as shown in Figure 5.22,

- (i) a' is contained in $B_r(p)$, that is $\|p - a'\| < r$.
- (ii) a' is outside $B_r(p)$, that is $\|p - a'\| > r$.

We start with case (i). We assume $\|p - a'\| < r = \|p - a\|$ and a' is the closest point on \mathbb{X} to a . The line connecting p and a' is normal to T_a . We draw a ball of radius $r^* = \frac{\|p - a'\|}{2}$ centered at p^* , where p^* is the mid-point between p and a' . Then $B = B_{r^*}(p^*)$ is tangent to $T_{a'}$ and intersects \mathbb{X} in at least the one other point p . Hence we can, if necessary, shrink B , keeping it tangent to a' the entire time, until we obtain a ball which intersects \mathbb{X} tangentially at a' and one other point and contains no points from \mathbb{X} in its interior; The center of this ball will of course then be a point from \mathcal{M} , and so we find $\tau \leq \text{lfs}(a') \leq r^* = \frac{\|p - a'\|}{2} < \frac{r}{2}$, which contradicts with our assumption that $r < \tau$.

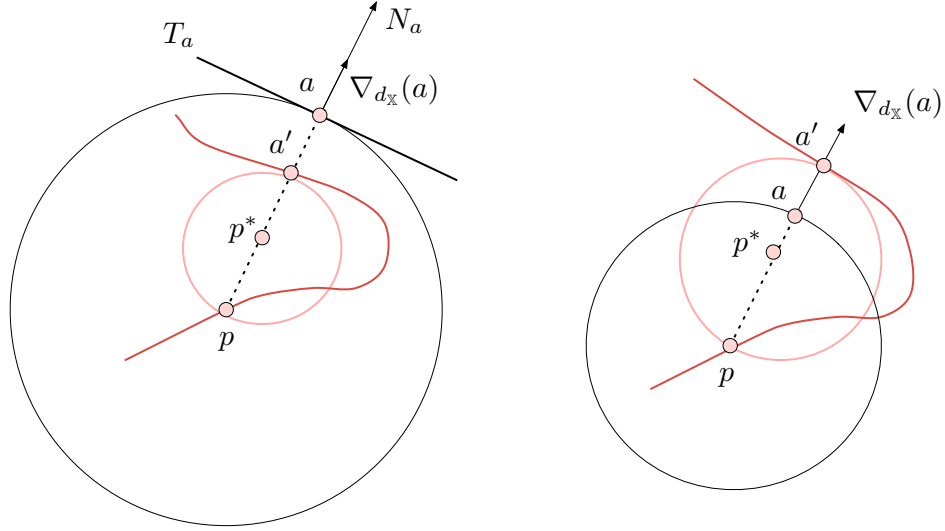


FIGURE 5.22: Left: case (i). Right: case(ii)

For case (ii) we assume $\|p - a'\| > r$. On the other hand, we must also have $\|p - a'\| = \|p - a\| + \|a - a'\| \leq 2r$, since $\|a - a'\| \leq \|p - a\| = r$ by our initial assumption. Again we draw a ball of radius $r^* = \frac{\|p - a'\|}{2}$ centered at the mid-point p^* between p and a' . Repeating the argument above, we find $\tau \leq \text{lfs}(a') \leq r^* = \frac{\|p - a'\|}{2} \leq r$, a contradiction. \square

We prove Theorem **B.1**.

Proof. By applying Proposition **B.2** to any point that is in $G(p, r)$, $G(q, r)$ or $G(p, q, r)$, we conclude that if $r < \tau$, then $\eta(p, r) = \eta(q, r) = \eta(p, q, r) = r$. \square

C Algorithm Details

We describe the details in constructing the simplicial complexes described in our algorithm. The various simplicial complexes, L , L_0 , K and K_0 , are the nerves of collections of convex sets. Here we go through the construction of L , constructions of the others follow similarly.

Implicit Perturbations. A direct approach to constructing L , the nerve of the collection \mathcal{A} , runs into difficulties as the corners of the convex sets created by the hyperplane P can be shared by many sets. To cope with this difficulty, we perturb these convex sets ever so slightly so that they meet in general positions. Note that this is not done by perturbing the hyperplane but rather decomposing it into pieces.

We are interested in the restricted Voronoi diagram of the sublevel sets inside the ball $B_r(p)$, which we denote as $\mathcal{V} = \text{Voi}(U|U_\alpha \cap B_r(p))$. The *restricted Voronoi cell* of u_i is defined as $V(u_i|U_\alpha \cap B_r(p)) = V(u_i) \cap B_r(p)$.

Given \mathcal{V} , we create three sets of points. Let \mathcal{T}_{pq} be the set of points $u_i \in U$ such that its restricted Voronoi cell intersects with the hyperplane, that is, $V(u_i|U_\alpha \cap B_r(p)) \cap P \neq \emptyset$. We impose an ordering of points in \mathcal{T}_{pq} , w.l.o.g., let the ordered set be $\mathcal{T}_{pq} = \{x_1, x_2, \dots, x_m\}$. \mathcal{T}_p is the set of points $u_i \in U$, such that $u_i \notin \mathcal{T}_{pq}$ and are closer to p (than to q). \mathcal{T}_q is the set of points $u_i \in U$, such that $u_i \notin \mathcal{T}_{pq}$ and are closer to q .

The hyperplane P intersects the restricted Voronoi cells of points in \mathcal{T}_{pq} . We denote these corresponding intersections as $\{P_1, P_2, \dots, P_m\}$. We perturb each P_i slightly such that no two pieces are colinear. Note that P_i is perpendicular to the direction $q - p$. A particular perturbation moves each P_i within the restricted Voronoi cell along the direction $q - p$ for $i\epsilon$ distance, where ϵ is sufficiently small. An example in \mathbb{R}^2 is shown in Figure 5.23.

Given such a perturbation, we call $\tilde{\mathcal{A}}$ the collection of *perturbed convex sets* and compute $\tilde{L} = \text{Nerve}(\tilde{\mathcal{A}})$ instead of $L = \text{Nerve}(\mathcal{A})$. By the properties of nerve construction, $\text{Nerve}(\tilde{\mathcal{A}}) \simeq \bigcup \tilde{\mathcal{A}}$, $\text{Nerve}(\mathcal{A}) \simeq \bigcup \mathcal{A}$. Since $\bigcup \tilde{\mathcal{A}} = \bigcup \mathcal{A}$, then we have $\tilde{L} \simeq L$. Now we describe how we construct \tilde{L} .

Case Analysis. Let L' be the restricted Delaunay triangulation, $L' = \text{Del}(U|U_\alpha \cap B_r(p))$. We read the simplicies from L' without explicit perturbations. Specifically, we follow a set of rules as follows to construct \tilde{L} from L' .

Since the hyperplane divides the restricted Voronoi cell of a point $x \in \mathcal{T}_{pq}$ into two convex sets, let x^p represent the perturbed convex set closer to p in the nerve construction, and let x^q represent the other set. Let σ be a simplex in L' with k vertices, that is, $\sigma = [y_1, y_2, \dots, y_k]$. There are seven cases regarding the membership of the points $\{y_1, y_2, \dots, y_k\}$.

1. All $y_i \in \sigma$ belong to \mathcal{T}_p . We add simplex $[y_1, y_2, \dots, y_k]$ to \tilde{L} .
2. All $y_i \in \sigma$ belong to \mathcal{T}_q . Same as case 1. We add simplex $[y_1, y_2, \dots, y_k]$ to \tilde{L} .
3. All $y_i \in \sigma$ belong to \mathcal{T}_{pq} . Suppose $\{y_1, y_2, \dots, y_k\}$ are sorted according to the ordering in \mathcal{T}_{pq} . We add the following simplicies and their faces to \tilde{L} :

$$\begin{aligned}
& [y_1^p, \dots, y_m^p, y_1^q] \\
& [y_2^p, \dots, y_m^p, y_1^q, y_2^q] \\
& [y_3^p, \dots, y_m^p, y_1^q, y_2^q, y_3^q] \\
& \dots \\
& [y_m^p, y_1^q, y_2^q, \dots, y_m^q]
\end{aligned}$$

4. Some y_i are in \mathcal{T}_p , the rest are in \mathcal{T}_{pq} . Suppose $\{y_{i_1}, \dots, y_{i_n}\} \subseteq \mathcal{T}_p$ and $\{y_{j_1}, \dots, y_{j_l}\} \subseteq \mathcal{T}_{pq}$. We add $[y_{i_1}, \dots, y_{i_n}, y_{j_1}^p, \dots, y_{j_l}^p]$ to \tilde{L} .
5. Some y_i are in \mathcal{T}_q , the rest are in \mathcal{T}_{pq} . Similar to case 4, suppose $\{y_{i_1}, \dots, y_{i_n}\} \subseteq \mathcal{T}_q$ and $\{y_{j_1}, \dots, y_{j_l}\} \subseteq \mathcal{T}_{pq}$. We add $[y_{i_1}, \dots, y_{i_n}, y_{j_1}^q, \dots, y_{j_l}^q]$ to \tilde{L} .
6. Some y_i are in \mathcal{T}_p , the rest are in \mathcal{T}_q . We show that Case 6 is impossible. Let $y_i \in \mathcal{T}_p$ and $y_j \in \mathcal{T}_q$ such that y_i and y_j are connected by an edge. Since y_i and y_j are on the opposite sides of P , the edge must intersect P at a point z . Then their corresponding restricted Voronoi cells, $V(y_i|U_\alpha \cap B_r(p))$ and $V(y_j|U_\alpha \cap B_r(p))$, must meet at a Voronoi face, which contains the point z . Suppose that the Voronoi

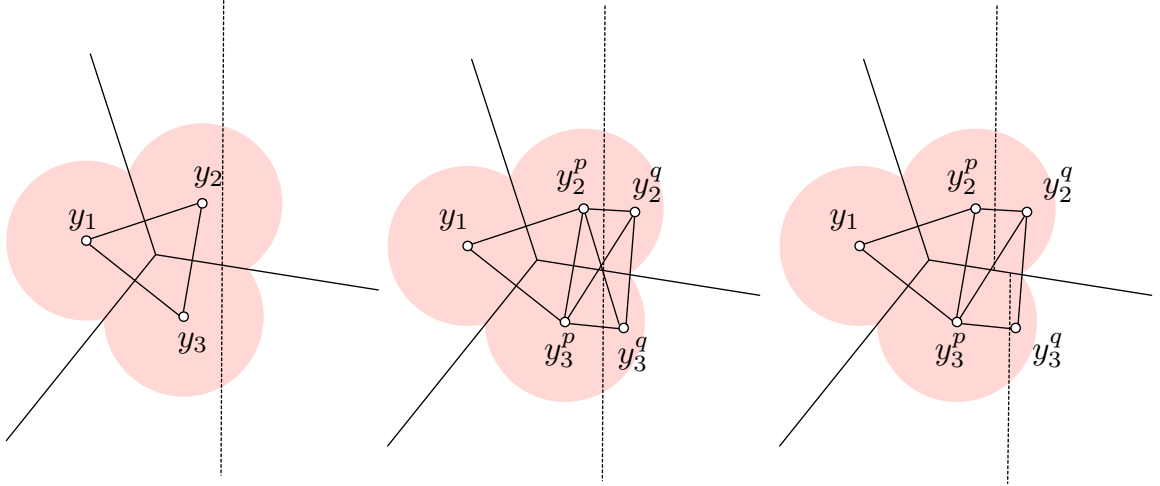


FIGURE 5.23: An example of the implicit perturbation. Dotted lines are the hyperplanes. A simplex $[y_1, y_2, y_3] \in L'$ is shown in the left. The simplices in L and in \tilde{L} are shown in the middle and right, respectively.

face is in general position, that is, it is not parallel to P . Then P intersects the Voronoi face, by definition, y_i and y_j must belong to \mathcal{T}_{pq} . This is a contradiction.

7. Some y_i are in \mathcal{T}_p , some are in \mathcal{T}_q , and the rest are in \mathcal{T}_{pq} . We show that case 7 is impossible using the same proof in case 6.

A simple example is shown in Figure 5.23. Given $[y_1, y_2, y_3] \in L'$, simplex $[y_1, y_2^p, y_3^p]$ is added to \tilde{L} according to case 4. Given $[y_2, y_3] \in L'$, simplices $[y_2^p, y_3^p, y_2^q]$, $[y_3^p, y_2^q, y_3^q]$ and their faces are added to \tilde{L} according to case 3.

In summary, we construct \tilde{L} by iterating through all simplices σ in L' , adding new simplicies to \tilde{L} constructed from σ following the above cases.

D Algorithmic Correctness

We prove the correctness of the algorithm described in Section 5.6.2 by proving Theorem 5.6.5. More precisely, we will prove that diagram 5.6 commutes, with the vertical arrows being isomorphisms, for some arbitrary but fixed choices of $\alpha < \beta$; we will omit the very similar argument about cokernels. The proof is unfortunately lengthy, and at times a

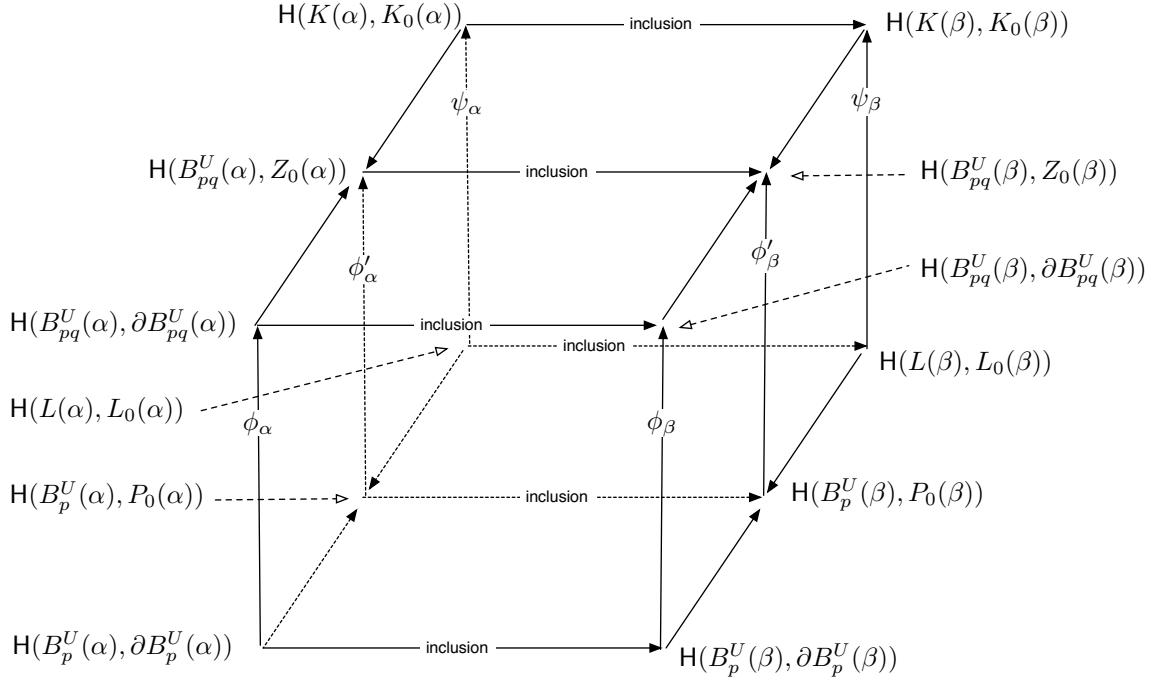


FIGURE 5.24: Two adjacent commuting cubes.

bit technical, for in order to prove our statements about diagram 5.6, we must also prove similar statements about several other interlocking diagrams. For sanity and clarity of presentation, we first exhibit all the diagrams at once in the form of the following double-cube (Figure 5.24).

D.1 Bottom Face

The bottom face of the double-cube has been detached and drawn in diagram 5.7. The horizontal maps in the upper square are induced by inclusion of pairs, and so the upper

square certainly commutes.

$$\begin{array}{ccc}
\mathrm{H}(B_p^U(\alpha), \partial B_p^U(\alpha)) & \xrightarrow{j_\alpha^\beta} & \mathrm{H}(B_p^U(\beta), \partial B_p^U(\beta)) \\
\downarrow i_\alpha & & \downarrow i_\beta \\
\mathrm{H}(B_p^U(\alpha), P_0(\alpha)) & \xrightarrow{j_\alpha^\beta} & \mathrm{H}(B_p^U(\beta), P_0(\beta)) \\
\uparrow h_\alpha & & \uparrow h_\beta \\
\mathrm{H}(L(\alpha), L_0(\alpha)) & \xrightarrow{g_\alpha^\beta} & \mathrm{H}(L(\beta), L_0(\beta)).
\end{array} \tag{5.7}$$

We have already shown that the two vertical maps in the upper square are isomorphisms; this was the content of the Power Cell Lemma in Section 5.6.2. To show that the vertical maps in the lower square are isomorphisms requires a bit more work. We make use of the following lemma, proven in [20].

Lemma D.1 (General Nerve Subdivision Lemma (GNSL)). *Let \mathcal{C} be the collection of maximal cells of a CW complex, each a convex set in \mathbb{R}^k . Define $f : |\mathrm{Sd} N| \rightarrow \cup \mathcal{C}$ by piecewise linear interpolation of its values at the vertices. If $f(\hat{\sigma})$ is contained in the intersection of the cells that correspond to the vertices of σ , for each simplex $\sigma \in N$, then f is a homotopy equivalence.*

The vertical isomorphisms in the bottom square then follow from this next lemma, where we may of course replace α with β if we wish.

Lemma D.2 (Nerve Subdivision Lemma). *Define $h = h_\alpha : |\mathrm{Sd} L(\alpha)| \rightarrow B_p^U(\alpha)$ on the vertices $\hat{\sigma}$ of $\mathrm{Sd} L(\alpha)$ by the formula*

$$h_\alpha(\hat{\sigma}) = \arg \min_{x \in V^\sigma \cap U_\alpha \cap B_r(p)} d_U^2(x) - d_p^2(x),$$

and extend it by linear interpolation. Then h_α is a homotopy equivalence of pairs from $(|\mathrm{Sd} L(\alpha)|, |\mathrm{Sd} L_0(\alpha)|)$ to $(B_p^U(\alpha), P_0(\alpha))$.

Proof. By construction, $h(\hat{\sigma})$ is contained in the intersection of the cells that correspond to the vertices of σ . By the GNSL then, h is a homotopy equivalence.

Now we need to prove that the restriction of h to $\text{Sd } L_0(\alpha)$ is also a homotopy equivalence. Let $\sigma \in L_0(\alpha)$ and put $h(\hat{\sigma}) = z$. For purposes of contradiction, suppose $z \notin P_0(\alpha)$. This means that $z \in \text{int } P(\alpha)$, by definition, and hence $d_U(z)^2 - d_p(z)^2 > \alpha^2 - r^2$.

Now choose some $z' \in V^\sigma \cap P_0$, which must exist since $\sigma \in L_0(\alpha)$. Then by definition we have $d_p(z')^2 - r^2 \geq d_U(z')^2 - \alpha^2$, or $d_U(z')^2 - d_p(z')^2 \leq \alpha^2 - r^2$. Combining the above inequalities, we have $d_U(z')^2 - d_p(z')^2 \leq \alpha^2 - r^2 < d_U(z)^2 - d_p(z)^2$, which contradicts the assumption that $h(\hat{\sigma}) = z$. We conclude that $z \in V^\sigma \cap P_0(\alpha)$. Applying the GNSL once more finishes the proof. \square

To show that the lower square commutes, we put $e = j_\alpha^\beta \circ h_\alpha$ and $e' = h_\beta \circ g_\alpha^\beta$, and we consider the map $H : |L(\alpha)| \times [0, 1] \rightarrow U_\alpha \cap B_r(p)$ defined by $H(x, t) = h_{\alpha_t} \circ g_\alpha^{\alpha_t}(x)$, where $\alpha_t = (1 - t)\alpha + t\beta$. Since the maps g and j are inclusions and the maps h vary continuously with α , H is a homotopy between e and e' . This implies that the induced homomorphisms between the corresponding homology groups are the same, $e_* = e'_*$.

D.2 Top Face

We detach the top face of Figure 5.24, drawing it in diagram 5.8. As before, we prove that all vertical maps are isomorphisms. The commutativity of the two smaller squares follows from nearly identical arguments to the ones used for the bottom face.

$$\begin{array}{ccc}
\mathrm{H}(B_{pq}^U(\alpha), \partial B_{pq}^U(\alpha)) & \rightarrow & \mathrm{H}(B_{pq}^U(\beta), \partial B_{pq}^U(\beta)) \\
\downarrow i'_\alpha & & \downarrow i'_\beta \\
\mathrm{H}(B_{pq}^U(\alpha), Z_0(\alpha)) & \rightarrow & \mathrm{H}(B_{pq}^U(\beta), Z_0(\beta)) \\
\uparrow h'_\alpha & & \uparrow h'_\beta \\
\mathrm{H}(K(\alpha), K_0(\alpha)) & \rightarrow & \mathrm{H}(K(\beta), K_0(\beta)) \tag{5.8}
\end{array}$$

The Intersection Power Cell Lemma tells us that the vertical maps in the top square are isomorphisms. As promised, we give the proof of this lemma here, repeating the statement for completeness.

Lemma D.3 (Intersection Power Cell Lemma). *Assume $B_r(p) \cap B_r(q) - Z_0(\alpha) \neq \emptyset$. The identity i' on $B_{pq}^U(\alpha)$ is a homotopy equivalence of pairs between $(B_{pq}^U(\alpha), \partial B_{pq}^U(\alpha))$ and $(B_{pq}^U(\alpha), Z_0(\alpha))$.*

Proof. It suffices to show that the restriction of the identity, $i' = i'_\alpha : \partial B_{pq}^U(\alpha) \rightarrow Z_0(\alpha)$, is a homotopy equivalence. To do this, we first define a retraction $j : Z_0(\alpha) \rightarrow \partial B_{pq}^U(\alpha)$ as follows. Fix a point $y \in \text{int } Z(\alpha)$, recalling that this set is nonempty by assumption. For each point $x \in Z_0(\alpha)$, we consider the unique ray starting at y and passing through x , and we let $x' = j(x)$ denote its intersection with $\partial B_{pq}^U(\alpha)$. Note that $x' \in Z_0(\alpha) \subseteq U(\alpha)$, and so j is certainly well-defined. That j is a retraction, meaning $j \circ i'$ is the identity on $\partial B_{pq}^U(\alpha)$, is obvious. On the other hand, the map

$$\lambda : Z_0(\alpha) \times [0, 1] \rightarrow Z_0(\alpha)$$

defined by $\lambda(x, t) = (1 - t)x + tx'$ is a homotopy between $i' \circ j$ and the identity map on $Z_0(\alpha)$, and so the claim follows. \square

To prove that the vertical maps in the lower square are isomorphisms, we again make use of the GNSL.

Lemma D.4 (Intersection Nerve Subdivision Lemma (INSL)). *Define*

$$h' = h'_\alpha : |\text{Sd } K(\alpha)| \rightarrow B_{pq}^U(\alpha)$$

by setting

$$h'_\alpha(\hat{\sigma}) = \arg \min_{x \in V^\sigma \cap U_\alpha \cap B_r(p) \cap B_r(q)} \min\{d_U^2(x) - d_p^2(x), d_U^2(x) - d_q^2(x)\},$$

where $\hat{\sigma}$ is the barycentre of $\sigma \in K(\alpha)$, and then extending it by linear interpolation. Then h' is a homotopy equivalence of pairs between $(|\text{Sd } K(\alpha)|, |\text{Sd } K_0(\alpha)|)$ and $(B_{pq}^U(\alpha), Z_0(\alpha))$.

Proof. The proof is quite similar to that of the NSL. By construction, $h'(\hat{\sigma})$ is contained in the intersection of the cells that correspond to the vertices of σ , and so we need only prove that the restriction of h' to the barycentric subdivision of $K_0(\alpha)$ is also a homotopy equivalence. Let $\sigma \in K_0(\alpha)$ and put $h'(\hat{\sigma}) = z$.

Suppose $z \notin Z_0(\alpha)$, and thus $z \in \text{int } Z(\alpha)$. By definition then, $d_p(z)^2 - r^2 < d_U(z)^2 - \alpha^2$ and $d_q(z)^2 - r^2 < d_U(z)^2 - \alpha^2$. In other words, $\min\{d_U^2(z) - d_p^2(z), d_U^2(z) - d_q^2(z)\} > \alpha^2 - r^2$.

Choose some point $z' \in V^\sigma \cap Z_0(\alpha)$. Then one of the following inequalities must hold: $d_p(z')^2 - r^2 \geq d_U(z')^2 - \alpha^2$, or $d_q(z')^2 - r^2 \geq d_U(z')^2 - \alpha^2$. That is, $\min\{d_U^2(z') - d_p^2(z'), d_U^2(z') - d_q^2(z')\} \leq \alpha^2 - r^2$.

Therefore, combining both inequalities, $\min\{d_U^2(z') - d_p^2(z'), d_U^2(z') - d_q^2(z')\} \leq \alpha^2 - r^2 < \min\{d_U^2(z) - d_p^2(z), d_U^2(z) - d_q^2(z)\}$, which contradicts the definition of z . \square

D.3 Left and Right Faces

$$\begin{array}{ccc}
\mathrm{H}(B_p^U, \partial B_p^U) & \xrightarrow{\phi} & \mathrm{H}(B_{pq}^U, \partial B_{pq}^U) \\
\downarrow i_* & & \downarrow i'_* \\
\mathrm{H}(B_p^U, P_0) & \xrightarrow{\phi'} & \mathrm{H}(B_{pq}^U, Z_0) \\
\uparrow h_* & & \uparrow h'_* \\
\mathrm{H}(L, L_0) & \xrightarrow{\psi} & \mathrm{H}(K, K_0).
\end{array} \tag{5.9}$$

We now come to the final and most complicated part of the correctness proof, involving the left face (diagram 5.9) of the double-cube; of course, everything we prove here will also hold for the right face. We have already established that all vertical maps are

isomorphisms, and now must show that both squares commute. The top square will in fact commute even on the chain level. The bottom square is a little more complicated, and we start by addressing this first.

In diagram 5.10, this bottom square has been expanded into two smaller squares of chain groups connected by chain maps. We show that the lower of these squares commutes on the chain level, and that the two choices of path across the upper square are connected by a chain homotopy.

$$\begin{array}{ccc}
C(B_p^U, P_0) & \xrightarrow{j'} & C(B_{pq}^U, Z_0) \\
\uparrow h_{\#} & & \uparrow h'_{\#} \\
C(|\text{Sd } L|, |\text{Sd } L_0|) & \xrightarrow{f'} & C(|\text{Sd } K|, |\text{Sd } K_0|). \\
\uparrow \eta & & \uparrow \eta \\
C(\text{Sd } L, \text{Sd } L_0) & \xrightarrow{f} & C(\text{Sd } K, \text{Sd } K_0). \tag{5.10}
\end{array}$$

Map Details

First we need to discuss two of the horizontal chain maps from diagram 5.10 in more explicit detail.

Upper map. We analyze the effect of j' on an arbitrary linear singular simplex $\omega : \Delta_p \rightarrow B_p^U$, where $\omega = l(a_0, \dots, a_p)$ for some points a_i in Euclidean space. The analysis can be broken up into three main cases:

(A.1) $\omega(\Delta_p) \subseteq B_q^U$: Then j' maps ω through unchanged, meaning:

$$[\omega : \Delta_p \rightarrow B_p^U] \xrightarrow{j'} [\omega : \Delta_p \rightarrow B_{pq}^U].$$

From now on we simplify notation by omitting the domain and range of the singular simplex, writing instead: $\omega \xrightarrow{j'} \omega$.

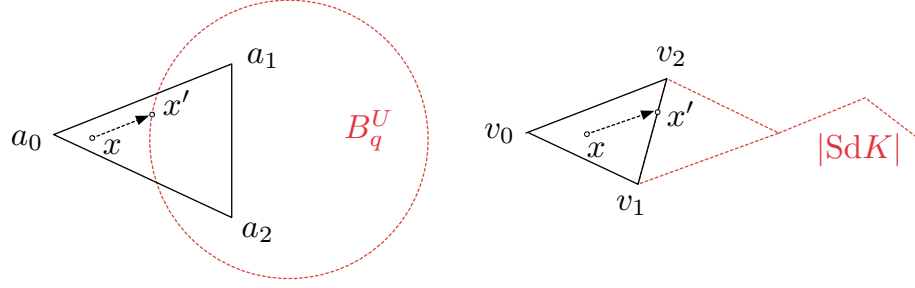


FIGURE 5.25: Left: map j' for a linear singular simplex $l(a_0, a_1, a_2)$. Right: map f' for a simplicial linear singular simplex $l(v_0, v_1, v_2)$.

(A.2) $\omega(\Delta_p) \cap B_q^U = \emptyset$: Then $\omega \xrightarrow{j'} 0$.

(A.3) $\omega(\Delta_p) \not\subseteq B_q^U$ and $\omega(\Delta_p) \cap B_q^U \neq \emptyset$. Here we have two sub-cases:

(A.3.a) ω is \mathcal{A}' -small: This implies that $\omega(\Delta_p) \subseteq X - V$. Map j' can be interpreted as a retraction. That is, letting $T = \omega(\Delta_p)$, $S = \omega(\Delta_p) \cap B_q^U$ and $R = \omega(\Delta_p) \cap \partial B_q^U$, we define $r : T \rightarrow S$ by: for $x \in S$, $r(x) = x$; for $x \in T - S$, $r(x) = x'$, where $x' \in R$, as shown in the left of Figure 5.25. Then $\omega \xrightarrow{j'} \tau$, where $\tau : \Delta_p \rightarrow B_{pq}^U$ is defined by: for $\epsilon \in \Delta_p$ where $\omega(\epsilon) \in S$, $\tau(\epsilon) = \omega(\epsilon)$; otherwise for $\epsilon \in \Delta_p$ where $\omega(\epsilon) \notin S$, $\tau(\epsilon) = r \circ \omega(\epsilon)$.

(A.3.b) ω is not \mathcal{A}' -small: We barycentrically subdivide ω enough times m until $\text{Sd}^m \omega$ is a \mathcal{A}' -small singular chain. Then each \mathcal{A}' -small singular simplex in $\text{Sd}^m \omega$ that has its image in $X - V$ follows the pattern of (A.3.a), resulting in a singular simplex $\tau_i : \Delta_p \rightarrow B_{pq}^U$. In the end we have, $\omega \xrightarrow{j'} c_\tau$, where c_τ is the singular chain, $c_\tau = \sum \tau_i$. This is shown in Figure 5.26.

Middle map. We now describe the action of f' on an arbitrary simplicial linear singular simplex. Let $\delta : \Delta_p \rightarrow |\text{Sd} L|$ be such a simplex with $\delta = \eta(\sigma) = l(v_0, \dots, v_p)$, for some simplex $\sigma = [v_0, \dots, v_p] \in \text{Sd} K$. As above, we have three cases to consider::

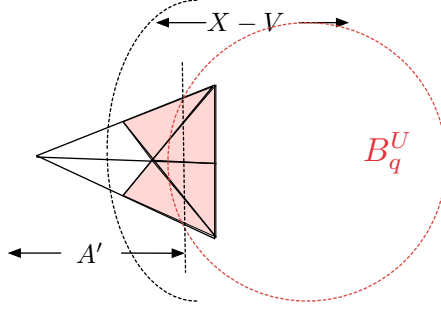


FIGURE 5.26: Map j' for a linear singular simplex that requires barycentric subdivision. In this illustrated example, all four shaded regions are the images of the four singular simplexes in the first barycentric subdivision which are \mathcal{A}' -small and have their images in $X - V$. Their formal sum gives a singular chain in $X - V$. Their retraction result in a singular chain in B_{pq}^U .

$$(B.1) \quad \delta(\Delta_p) \subseteq |\text{Sd } K|: \text{ then } \delta \xrightarrow{f'} \delta.$$

$$(B.2) \quad \delta(\Delta_p) \cap |\text{Sd } K| = \emptyset: \delta \xrightarrow{f'} 0.$$

(B.3) $\delta(\Delta_p) \not\subseteq |\text{Sd } K|$ and $\delta(\Delta_p) \cap |\text{Sd } K| \neq \emptyset$: From Lemma D.5 below, we know that

$$(\delta(\Delta_p) \cap |\text{Sd } K|) \subseteq |\text{Sd } K_0|, \text{ and so } \delta \xrightarrow{f'} 0.$$

Lemma D.5. *Given a simplex $\sigma \in L$, if $\sigma \notin K$ and there exists $\tau < \sigma$ such that $\tau \in K$, then $\tau \in K_0$.*

Proof. Suppose there exists $\omega < \tau$ such that $\omega \in K - K_0$. This implies that V^ω is completely contained in B_{pq}^U . Since V^σ is the intersection of V^ω with the partial Voronoi cells of vertices in σ that are not in ω , then V^σ should be completely contained in B_{pq}^U .

This means that σ is in K , which leads to a contradiction. \square

Lower Square

As promised, we now show that the lower square in diagram 5.10 commutes. Choose an arbitrary $\sigma = [v_0, \dots, v_p] \in \text{Sd } L$, where each v_i is a barycenter of some simplex σ' in L ; as always, we assume that the vertices are ordered by increasing dimension of their defining simplices. We have two cases:

(C.1) $\sigma \in \text{Sd } K$: by definition, $\eta(\sigma) = l(v_0, \dots, v_p)$ has its image in $|\text{Sd } K|$, and f is the identity map, that is, $\sigma \xrightarrow{f} \sigma \xrightarrow{\eta} \eta(\sigma)$. Meanwhile, by case (B.1), $\sigma \xrightarrow{\eta} \eta(\sigma) \xrightarrow{f'} \eta(\sigma)$. Therefore $(\eta \circ f)(\sigma) = (f' \circ \eta)(\sigma)$.

(C.2) $\sigma \notin \text{Sd } K$: then $\sigma \xrightarrow{f} 0 \xrightarrow{\eta} 0$. On the other had, since $\sigma \notin \text{Sd } K$, we know that the image of $\delta = \eta(\sigma) = l(v_0, \dots, v_p)$ cannot be entirely contained within $|\text{Sd } K|$. There are then two sub-cases to consider:

(C.2.a) $\delta(\Delta_p) \cap |\text{Sd } K| = \emptyset$: this is case (B.2). We have $\sigma \xrightarrow{\eta} \delta \xrightarrow{f'} 0$.

(C.2.b) $\delta(\Delta_p) \cap |\text{Sd } K| \subseteq |\text{Sd } K_0|$: this is case (B.3). We have $\sigma \xrightarrow{\eta} \delta \xrightarrow{f'} 0$.

Upper Square

Finally, we show that the upper square in diagram 5.10 commutes up to chain homotopy; that is, we will construct a chain homotopy D between the two chain maps $e = j' \circ h_{\#}$ and $e' = h'_{\#} \circ f'$. This will of course imply that $e_* = e'_*$; in other words, that the induced homology diagram commutes. For clarity, we zoom in on diagram 5.10 and draw the relevant portion below as diagram 5.11.

$$\begin{array}{ccc}
 C(B_p^U, P_0) & \xrightarrow{j'} & C(B_{pq}^U, Z_0) \\
 \uparrow h_{\#} & & \uparrow h'_{\#} \\
 C(|\text{Sd } L|, |\text{Sd } L_0|) & \xrightarrow{f'} & C(|\text{Sd } K|, |\text{Sd } K_0|).
 \end{array} \tag{5.11}$$

For notational ease, we set $X = |\text{Sd } L|$ and $Y = B_{pq}^U$. To construct D , we will define for each p a chain map $F_p : C_p(X \times I) \rightarrow C_p(Y)$, and then we will set $D_p = F_{p+1} \circ G_p$, where $G_p : C_p(X \times I) \rightarrow C_{p+1}(X \times I)$ is given by Lemma D.6 below.

Construction of F. Let $\pi : X \times I \rightarrow X$ be projection on the first factor, and fix an arbitrary simplicial linear singular simplex $\kappa : \Delta_p \rightarrow X \times I$. Then $\pi_{\#}(\kappa) = \delta = l(\hat{\sigma}_0, \dots, \hat{\sigma}_p)$,

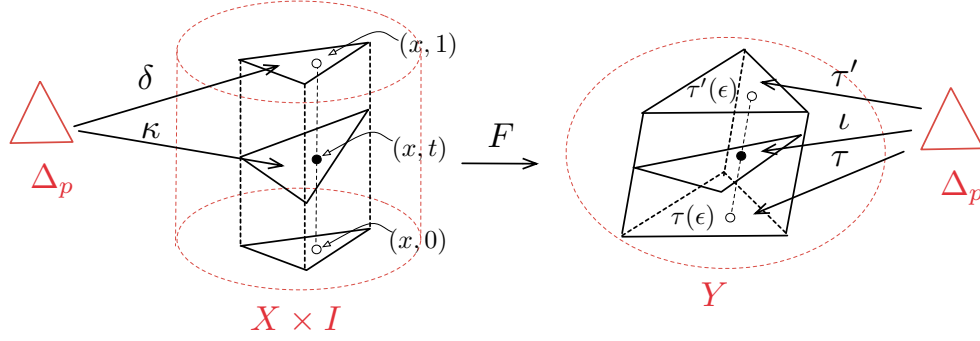


FIGURE 5.27: Case (D.1.a): illustration of F .

for some simplex $\sigma = [\hat{\sigma}_0, \dots, \hat{\sigma}_p]$ in $\text{Sd } L$. We define F in stages, based on properties of δ , as follows.

(D.1) $\delta(\Delta_p) \subseteq |\text{Sd } K|$: following the e' -path and case (B.1), we have $\delta \xrightarrow{j'} \delta \xrightarrow{h'_\#} \tau'$.

On the other hand, following the e -path results in $\delta \xrightarrow{h_\#} \omega$. We now have three sub-cases, based on varying properties of ω :

(D.1.a) $\omega(\Delta_p) \subseteq B_q^U$: following the e -path and case (A.1). we have, $\delta \xrightarrow{h_\#} \omega \xrightarrow{j'} \tau$, where $\tau = \omega$ except for differing range. We then can define $F(\kappa) = \iota$, where $\iota : \Delta_p \rightarrow Y$ is given by: for every $\epsilon \in \Delta_p$, where $\kappa(\epsilon) = (x, t) \in X \times I$, $\iota(\epsilon) = (1 - t)\tau(\epsilon) + t\tau'(\epsilon)$. This formula is illustrated in Figure 5.27.

(D.1.b) $\omega(\Delta_p) \cap B_q^U = \emptyset$: This is case (A.2). We branch further as follows:

(i) $\delta(\Delta_p) \subseteq |\text{Sd } K_0|$: Following the e' -path, $\delta \xrightarrow{j'} 0 \xrightarrow{h'_\#} 0$. Similarly, following the e -path, $\delta \xrightarrow{h_\#} \omega \xrightarrow{j'} 0$. We define $F(\kappa) = 0$.

(ii) $\delta(\Delta_p) \not\subseteq |\text{Sd } K_0|$: this is not possible. Suppose it were. This implies that there exists at least one vertex $\hat{\sigma}_i$ of σ such that $V^{\sigma_i} \cap B_{pq}^U \neq \emptyset$ and $V^{\sigma_i} \cap Z_0 = \emptyset$. This means that V^{σ_i} is completely contained in $B_r(q)$. Therefore $h(\hat{\sigma}_i)$ is contained in $B_r(q)$, which contradicts our assumption.

(D.1.c) $\omega(\Delta_p) \not\subseteq B_q^U$ and $\omega(\Delta_p) \cap B_q^U \neq \emptyset$: we must consider two further sub-cases.

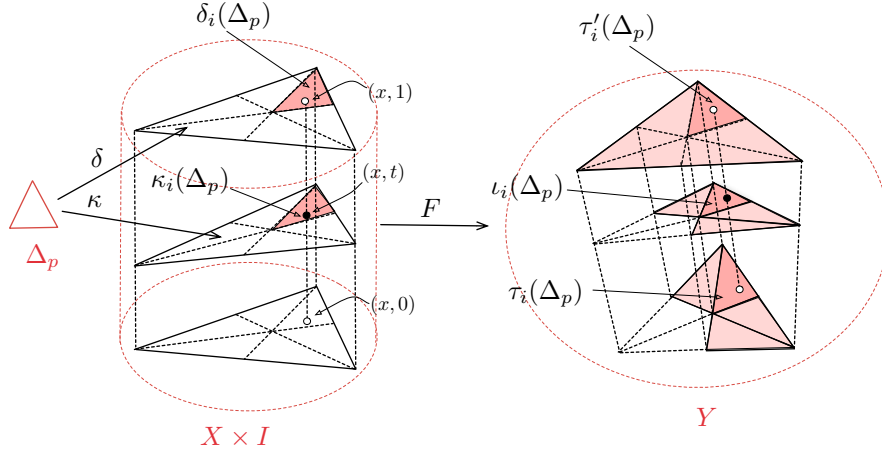


FIGURE 5.28: Case (D.1.c): illustration of F . Left: the dark shaded region is the minimal carrier of δ_i and κ_i . Right: the shaded regions from top to bottom are the minimal carriers of τ'_i , c_ι and c_τ respectively; the dark shaded regions from top to bottom are the minimal carriers of τ'_i , l_i and τ_i , respectively. For simplicity, we illustrate the minimal carrier of the singular chain c_τ as the union of the minimal carriers of its simplexes before their retraction.

- (i) ω is \mathcal{A}' -small: this is case (A.3.a), and we define $F(\kappa)$ similarly to case (D.1.a).
- (ii) ω is not \mathcal{A}' -small: this is case (A.3.b). Then $\delta \xrightarrow{h_\#} \omega \xrightarrow{j'} c_\tau$, where $c_\tau = \sum \tau_i$ for some collection of $\tau_i : \Delta_p \rightarrow B_{pq}^U$. We now define $F(\kappa) = c_\iota$, where $c_\iota = \sum \iota_i$ and each singular simplex $\iota_i : \Delta_p \rightarrow Y$ is defined as follows. Let m be the smallest integer such that $\text{Sd}^m \omega$ is \mathcal{A}' -small. For each singular simplex τ_i in c_τ , there exists a singular simplex ω_i in $\text{Sd}^m \omega$ such that $j'(\omega_i) = \tau_i$. For each such ω_i , there exists a singular simplex δ_i in $\text{Sd}^m \delta$ such that $h_\#(\delta_i) = \omega_i$. In other words, for each τ_i in c_τ , there exist δ_i in $\text{Sd}^m \delta$, such that following the e -path, $\delta_i \xrightarrow{h_\#} \omega_i \xrightarrow{j'} \tau_i$. Meanwhile, for each such δ_i , following the e' -path gives $\delta_i \xrightarrow{f'} \delta_i \xrightarrow{h'_\#} \tau'_i$. On the other hand, for each such δ_i , there exists a corresponding κ_i in $\text{Sd}^m \kappa$, such that $\delta_i = \pi(\kappa_i)$. We now define ι_i for each such κ_i . For all $\epsilon \in \Delta_p$ where $\kappa_i(\epsilon) = (x, t) \in X \times I$, $\iota_i(\epsilon) = (1 - t)\tau_i(\epsilon) + t\tau'_i(\epsilon)$. This is illustrated in Figure 5.28.

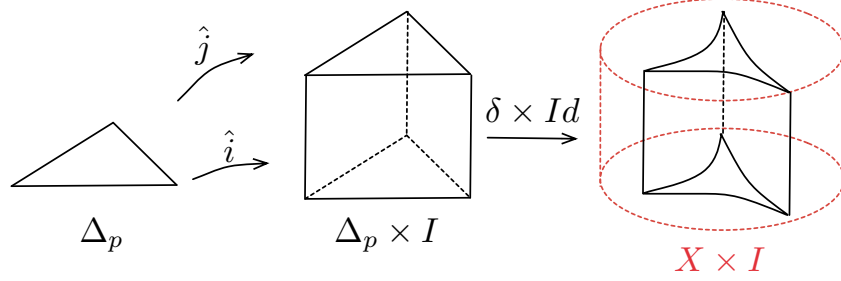


FIGURE 5.29: Illustration of G .

(D.2) $\delta(\Delta_p) \not\subseteq |\text{Sd } K|$: we again have two sub-cases:

(D.2.a) $\delta(\Delta_p) \cap |\text{Sd } K| = \emptyset$: following the e' -path and case (B.2), $\delta \xrightarrow{f'} 0 \xrightarrow{h'_\#} 0$. Since $\delta(\Delta_p) \cap |\text{Sd } K| = \emptyset$, this implies that its corresponding $\sigma \notin \text{Sd } K$. That is, for all $\hat{\sigma}_i$, $V^{\sigma_i} \cap B_{pq}^U = \emptyset$, therefore all $h(\hat{\sigma}_i)$ lie outside of $B_r(q)$. Let $\omega = h_\#(\delta) = h \circ \delta$. This means ω has its image outside of B_q^U . Then following the e -path, $\delta \xrightarrow{h_\#} \omega \xrightarrow{j'} 0$. We define $F(\kappa) = 0$.

(D.2.b) $(\delta(\Delta_p) \cap |\text{Sd } K|) \subseteq |\text{Sd } K_0|$: following the e' -path and case (B.3), we have, $\sigma \xrightarrow{f'} 0 \xrightarrow{h'_\#} 0$. On the other hand, let $\omega = h_\#(\delta) = h \circ \delta$. Then $\omega(\Delta_p) \subseteq P_0$ and so following the e -path give $\sigma \xrightarrow{h_\#} \omega \xrightarrow{j'} 0$. We define $F(\kappa) = 0$.

Construction of D. To define our chain homotopy D , we first need the following lemma:

Lemma D.6. ([82], page 171) *There exists for each space X and each non-negative integer p , a homomorphism $G_p : C_p(X) \rightarrow C_{p+1}(X \times I)$, having the following property: if $\delta : \Delta_p \rightarrow X$ is a singular simplex, then $\partial G\delta + G\partial\delta = j_\#(\delta) + i_\#(\delta)$, where the map $i : X \rightarrow X \times I$ carries x to $(x, 0)$, and the map $j : X \rightarrow X \times I$ carries x to $(x, 1)$.*

This homomorphism is illustrated intuitively in Figure 5.29, where $\delta \times Id$ carries a singular $p + 1$ chain that fills up the entire prism $\Delta_p \times I$ to a singular chain on $X \times I$, and the maps $\hat{i}, \hat{j} : \Delta_p \rightarrow \Delta_p \times I$ carry each x to $(x, 0)$ and $(x, 1)$ respectively. Then, as

promised, we set $D_p = F_{p+1} \circ G_p$. To show that D is a chain homotopy between e and e' , we calculate

$$\begin{aligned}
\partial D &= \partial(FG) \\
&= F\partial G \\
&= F(j_{\#} + i_{\#} + G\partial) \\
&= Fj_{\#} + Fi_{\#} + FG\partial \\
&= Fj_{\#} + Fi_{\#} + D\partial
\end{aligned}$$

Hence we need only show that $Fj_{\#} = e'$ and $Fi_{\#} = e$ to complete the argument. In the case when $F(\kappa)$ is defined to be 0, the corresponding $e(\delta)$ and $e'(\delta)$ are also 0, so this is no problem. In the case when $F(\kappa)$ is not defined to be 0, as shown in Figure 5.27 and Figure 5.28, $Fj_{\#}(\delta) = e'(\delta)$, and $Fi_{\#}(\delta) = e(\delta)$. This concludes the proof that the upper square in diagram 5.10 commutes up to chain homotopy, and thus that the bottom square of diagram 5.9 commutes.

Top Square of Diagram 5.9

As promised above, we now prove that the top square of diagram 5.9 commutes, which will complete the proof that the left face of Figure 5.24 commutes. In fact, the top square commutes on the chain level, which we draw directly below.

$$\begin{array}{ccc}
C(B_p^U, \partial B_p^U) & \xrightarrow{j} & C(B_{pq}^U, \partial B_{pq}^U) \\
\downarrow i_{\#} & & \downarrow i'_{\#} \\
C(B_p^U, P_0) & \xrightarrow{j'} & C(B_{pq}^U, Z_0)
\end{array}$$

Setting $e = j' \circ i_{\#}$ and $e' = i'_{\#} \circ j$, we show, once again via an exhaustive case analysis, that $e = e'$.

First we need to understand the map j for a linear singular simplex. The interpretation of j is almost the same as that of j' (case (A)). More specifically, we let $\omega : \Delta_p \rightarrow B_p^U$ be an arbitrary linear singular simplex. There are three cases:

(E.1) $\omega(\Delta_p) \subseteq B_q^U$: then $\omega \xrightarrow{j} \omega$.

(E.2) $\omega(\Delta_p) \cap B_q^U = \emptyset$: then $\omega \xrightarrow{j} 0$.

(E.3) $\omega(\Delta_p) \not\subseteq B_q^U$ and $\omega(\Delta_p) \cap B_q^U \neq \emptyset$: We have two sub-cases:

(E.3.a) ω is \mathcal{A} -small: then $\omega \xrightarrow{j} \gamma$, where $\gamma : \Delta_p \rightarrow B_{pq}^U$ is defined via the retraction-type arguments above.

(E.3.b) ω is not \mathcal{A} -small: then $\omega \xrightarrow{j'} c_\gamma$, where $c_\gamma = \sum \gamma_i$, with each $\gamma_i : \Delta_p \rightarrow B_{pq}^U$ described by the subdivision and retraction arguments we have already given.

To complete the proof, we fix an arbitrary singular simplex $\delta : \Delta_p \rightarrow B_p^U$, and again argue by cases.

(F.1) $\delta(\Delta_p) \subseteq B_q^U$: exploiting the analysis above, we note that following the e' -path results in $\delta \xrightarrow{j} \delta \xrightarrow{i'_\#} \delta$, while following the e -path gives $\delta \xrightarrow{i_\#} \delta \xrightarrow{j'} \delta$, as needed.

(F.2) $\delta(\Delta_p) \cap B_q^U = \emptyset$: here both paths result in 0.

(F.3) $\delta(\Delta_p) \not\subseteq B_q^U$ and $\delta(\Delta_p) \cap B_q^U \neq \emptyset$: here we must analyze two sub-cases:

(F.3.a) δ is \mathcal{A} -small: this implies that $\delta(\Delta_p) \subseteq X - V$. Following the e -path gives.

$\delta \xrightarrow{j} \gamma \xrightarrow{i'_\#} \gamma$. On the other hand, δ is also \mathcal{A}' -small, since \mathcal{A} and \mathcal{A}' share the element $X - V$, and hence the e' path

$$\delta \xrightarrow{j} \delta \xrightarrow{i'_\#} \tau.$$

But really the fact that $X - V$ is part of \mathcal{A} and \mathcal{A}' means that τ and γ follow the same retraction, and thus $\gamma = \tau$.

(F.3.b) δ is not \mathcal{A} -small: the analysis here is the same as the last case, with some words about subdivision added.

D.4 Finale

We are now ready to finish the proof of Theorem 5.6.5, which boils down to verifying that diagram 5.6 commutes, with the vertical maps being isomorphisms. That is,

$$\begin{array}{ccccc}
 \dots & \rightarrow & \ker \phi_\alpha^U & \rightarrow & \ker \phi_\beta^U & \rightarrow & \dots \\
 & & \uparrow \cong & & \uparrow \cong & & \\
 \dots & \rightarrow & \ker \psi_\alpha & \rightarrow & \ker \psi_\beta & \rightarrow & \dots
 \end{array}$$

But this is now just easy diagram-chasing. Commutativity of diagram 5.6 follows directly from the commutativity of the bottom face of the double-cube in Figure 5.24, and the leftmost (rightmost) vertical isomorphism derives from our statements about the left (right) face of the double-cube. The commutativity of the top face implies that the cokernel analogue to diagram 5.6 commutes, after a little more algebra which we omit.

Chapter 6

Discussion

In this thesis, we concern ourselves with feature extractions from several forms of data, graphs, triangulations and point cloud. We demonstrate that both the theory of persistent homology and statistics are effective in separating features from noise. Our work provide theoretical foundations for potential applications in computational biology, social science, visualization and machine learning.

We begin with Chapter 2 by generalizing spacial scan statistics from point sets to graphs, introducing the notion of *graph scan statistic* and its simplification, the *Poisson discrepancy*. The graph scan statistics infers graph clusterings by measuring the statistical significance of a densely-connected subgraph based on hypothesis testing and likelihood function. Since the graph scan statistics is used to detect locally best clusters, perhaps another interesting question is: can we derive a *global graph scan statistic* that is useful in finding the best partition of a graph? This is currently joint work with Randolph Rotta. We have some success defining such a global measure and relating it to *modularity*.

We gradually shift our focus from statistics to persistent homology, and from graphs to triangulations. In Chapter 3, we describe our first attempt in bringing statistical flavor to the theory of persistent homology by analyzing how noise influences summary statistics

on total persistence. We derive several theorems for PL functions defined on triangulations of topological spaces, relating the total persistence in expectation to the properties of the triangulation and the distribution. However, this is merely a simple test drive towards the direction of probabilistic topological analysis, by bridging the theory of statistics with persistent homology. This include providing sampling estimates for topological inference under different resolutions, quantify variance of persistence computation from noisy data, etc. One interesting question in this direction is, can we compute the notion of “average” persistence diagrams, from points sampled from manifolds, or from functions sampled from distributions?

We continue to explore feature extraction from triangulations in Chapter 4, specifically for triangulations of 2-manifolds. We use tools developed in persistent homology, the elevation function, to study features of triangulated protein surfaces. Since the elevation function is based on the persistence pairings of height functions, it is an interesting research direction to define the *general elevation function* based on various notions of distance functions, such as distances to a point, a line or a convex set.

Subsequently we work towards stratification learning from point cloud data in Chapter 5. We focus on inferring the local structure of a point in the sample in relation to its neighborhood. We examine this inference problem both theoretically and algorithmically. We provide inference statements from both topological and probabilistic perspectives. We give one result, the probabilistic inference theorem, in the direction of probabilistic topological analysis, namely, if we sample enough points i.i.d. uniform from the space, we can infer local structure with confidence. It will be interesting to apply our homological inference on real-world data, such as medical images and high-dimensional parameter spaces. Developing faster algorithms for stratification learning in practice is necessary. We envision our work to be applied by using computationally more convenient complexes such Rips or Witness complexes, or in combination with dimension reduction and dimension estimation techniques. Furthermore, we would like to improve the theoretical bounds,

make them more general under different models of uncertainty.

Bibliography

- [1] The biogeometry web-pages. <http://biogeometry.duke.edu>.
- [2] Computational geometry algorithms library. <http://www.cgal.org>.
- [3] Robert J. Adler, Omer Bobrowski, Matthew S. Borman, Eliran Subag, and Shmuel Weinberger. Persistent homology for random fields and complexes. Manuscript, 2010.
- [4] Deepak Agarwal, Andrew McGregor, Jeff M. Phillips, Suresh Venkatasubramanian, and Zhengyuan Zhu. Spatial scan statistics: approximations and performance study. In *Proceedings 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 24–33, 2006.
- [5] Deepak Agarwal, Jeff M. Phillips, and Suresh Venkatasubramanian. The hunting of the bump: on maximizing statistical discrepancy. In *Proceedings 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1137–1146, 2006.
- [6] Pankaj K. Agarwal, Herbert Edelsbrunner, John Harer, and Yusu Wang. Extreme elevation on a 2-manifold. *Discrete and Computational Geometry*, 36:553–572, 2006.
- [7] William Aiello, Fan Chung, and Linyuan Lu. Random evolution in massive graphs. *Handbook of massive data sets*, pages 97–122, 2002.
- [8] Réka Albert, Hawoong Jeong, and Albert-László Barabási. The diameter of the world wide web. *Nature*, 401:130–131, 1999.
- [9] Lyuba Alboul and Gilberto Echeverria. Polyhedral Gauss maps and curvature characterization of triangle meshes. *Lecture Notes in Computer Science*, 3605:14–33, 2005.
- [10] Uri Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. CRC Press, Boca Raton, FL, USA, 2007.

- [11] Gary D Bader and Christopher WV Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4, 2003.
- [12] Thomas F. Banchoff. Critical points and curvature for embedded polyhedral surfaces. *The American Mathematical Monthly*, 77:475–485, 1970.
- [13] Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, Srujana Merugu, and Dharmendra S. Modha. A generalized maximum entropy approach to bregman co-clustering and matrix approximation. *Journal of Machine Learning Research*, 8:1919–1986, 2007.
- [14] Arindam Banerjee, Srujana Merugu, Inderjit S. Dhillon, and Joydeep Ghosh. Clustering with bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005.
- [15] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
- [16] Jeffrey Baumes, Mark K. Goldberg, Mukkai S. Krishnamoorthy, Malik Magdonismail, and Nathan Preston. Finding communities by clustering a graph into overlapping subgraphs. In *Proceedings IADIS International Conference of Applied Computing*, pages 97–104, 2005.
- [17] M. J. Bayarri and M. H. DeGroot. Difficulties and ambiguities in the definition of a likelihood function. *Statistical Methods and Applications*, 1:1–15, 1992.
- [18] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396, 2002.
- [19] Paul Bendich. *Analyzing Stratified Spaces Using Persistent Versions of Intersection and Local Homology*. PhD thesis, Duke University, 2008.
- [20] Paul Bendich, David Cohen-Steiner, Herbert Edelsbrunner, John Harer, and Dmitriy Morozov. Inferring local homology from sampled stratified spaces. In *Proceedings 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 536–546, 2007.
- [21] Jean-Daniel Boissonnat, Olivier Devillers, and Samuel Hornus. Incremental construction of the delaunay triangulation and the delaunay graph in medium dimension. In *Proceedings 25th Annual Symposium on Computational Geometry*, pages 208–216, 2009.

- [22] Peter Bubenik, Gunnar Carlson, Peter T. Kim, and Zhi-Ming Luo. Statistical topology via morse theory, persistence and nonparametric estimation. *Contemporary Mathematics*, (to appear), 2010.
- [23] Peter Bubenik and Peter T. Kim. A statistical approach to persistent homology. *Homology, Homotopy and Applications*, 9:337–362, 2007.
- [24] George Casella and Roger L. Berger. *Statistical Inference*. Duxbury Press, Pacific Grove, CA, USA, 2002.
- [25] Frédéric Cazals, Frédéric Chazal, and Thomas Lewiner. Molecular shape analysis based upon the morse-smale complex and the connolly function. In *Proceedings 19th Annual Symposium on Computational Geometry*, pages 351–360, 2003.
- [26] Frédéric Chazal, David Cohen-Steiner, Marc Glisse, Leonidas J. Guibas, and Steve Y. Oudot. Proximity of persistence modules and their diagrams. In *Proceedings 25th Annual Symposium on Computational Geometry*, pages 237–246, 2009.
- [27] Frédéric Chazal, David Cohen-Steiner, and André Lieutier. A sampling theory for compact sets in euclidean space. *Discrete and Computational Geometry*, 41:461–479, 2009.
- [28] Frédéric Chazal and André Lieutier. Weak feature size and persistent homology: computing homology of solids in r^n from noisy data samples. In *Proceedings 21st Annual Symposium on Computational Geometry*, pages 255–262, 2005.
- [29] Ho-Lun Cheng, Tamal K. Dey, Herbert Edelsbrunner, and John Sullivan. Dynamic skin triangulation. *Discrete and Computational Geometry*, 25:525–568, 2001.
- [30] Fan Chung and Linyuan Lu. Connected components in random graphs with given expected degree sequences. *Annals of Combinatorics*, 6:125–145, 2002.
- [31] Kenneth L. Clarkson. Tighter bounds for random projections of manifolds. In *Proceedings 24th Annual Symposium on Computational Geometry*, pages 39–48, 2008.
- [32] Aaron Clauset. Finding local community structure in networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 72, 2005.
- [33] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70:066111, 2004.

- [34] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete and Computational Geometry*, 37:103–120, 2007.
- [35] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Extending persistence using Poincaré and Lefschetz duality. *Foundations of Computational Mathematics*, 9:79–103, 2009.
- [36] David Cohen-Steiner, Herbert Edelsbrunner, John Harer, and Yuriy Mileyko. Lipschitz functions have l_p -stable persistence. *Foundations of Computational Mathematics*, 10:127–139, 2010.
- [37] David Cohen-Steiner, Herbert Edelsbrunner, John Harer, and Dmitriy Morozov. Persistence homology for kernels, images and cokernels. In *Proceedings 20th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1011–1020, 2009.
- [38] David Cohen-Steiner, Herbert Edelsbrunner, and Dmitriy Morozov. Vines and vineyards by updating persistence in linear time. In *Proceedings 22nd Annual Symposium on Computational Geometry*, pages 119 – 126, 2006.
- [39] David Cohen-Steiner and Jean-Marie Morvan. Second fundamental measure of geometric sets and local approximation of curvatures. *Journal of Differential Geometry*, 74:363–394, 2006.
- [40] Kree Cole-McLaughlin, Herbert Edelsbrunner, John Harer, Vijay Natarajan, and Valerio Pascucci. Loops in Reeb graphs of 2-manifolds. *Discrete and Computational Geometry*, 32:231–244, 2004.
- [41] Michael L. Connolly. Analytical molecular surface calculation. *Journal of Applied Crystallography*, 6:548–558, 1983.
- [42] Michael L. Connolly. Shape complementarity at the hemoglobin $\alpha 1\beta 1$ subunit interface. *Biopolymers*, 25:1229–1247, 1986.
- [43] Marc de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf. *Computational Geometry - Algorithms and Applications*. Springer-Verlag, Berlin, Germany, 1997.
- [44] V. de Silva and G. Carlsson. Topological estimation using witness complexes. *Symposium on Point-Based Graphics*, pages 157–166, 2004.
- [45] Tamal K. Dey. *Curve and Surface Reconstruction*. Cambridge University Press, Cambridge, England, 2007.

- [46] Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis. Weighted graph cuts without eigenvectors: a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:1944–1957, 2007.
- [47] Inderjit S. Dhillon, Subramanyam Mallela, and Dharmendra S. Modha. Information-theoretic co-clustering. In *Proceedings 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 89–98, 2003.
- [48] Jordi Duch and Alex Arenas. Community detection in complex networks using extremal optimization. *Physical Review E*, 72:027104, 2005.
- [49] Herbert Edelsbrunner. Deformable smooth surface design. *Discrete and Computational Geometry*, 21:87–115, 1999.
- [50] Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. American Mathematical Society, Providence, RI, USA, 2010.
- [51] Herbert Edelsbrunner, David Letscher, and Afra J. Zomorodian. Topological persistence and simplification. *Discrete and Computational Geometry*, 28:511–533, 2002.
- [52] Samuel Eilenberg and Norman Steenrod. *Foundations of Algebraic Topology*. Princeton University Press, Princeton, NJ, USA, 1952.
- [53] Paul Erdős and Alfréd Rényi. On random graphs. i. *Publicationes Mathematicae*, 6:290–297, 1959.
- [54] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 104, pages 36–41, 2007.
- [55] Loukas Georgiadis, Robert E. Tarjan, and Renato F. Werneck. Design of data structures for mergeable trees. In *Proceedings 17th Annual ACM-SIAM Symposium on Discrete Algorithm*, pages 394–403, 2006.
- [56] Michelle Girvan and M. E. J. Newman. Community structure in social and biological networks. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 99, pages 7821–7826, 2002.
- [57] Joseph Glaz, Joseph I. Naus, and Sylvan Wallenstein. *Scan Statistics*. Springer-Verlag, New York, NY, USA, 2001.

- [58] Mark Goresky and Robert MacPherson. Intersection homology theory. *Topology*, 19:135–162, 1980.
- [59] Mark Goresky and Robert MacPherson. *Stratified Morse Theory*. Springer-Verlag, New York, NY, USA, 1988.
- [60] Marvin J. Greenberg and John R. Harper. *Algebraic Topology A First Course*. Addison-Wesley, Reading, MA, USA, 1981.
- [61] Gloria Haro, Gregory Randall, and Guillermo Sapiro. Stratification learning: Detecting mixed density and dimensionality in high dimensional point clouds. *Advances in NIPS*, 17, 2005.
- [62] H. Leon Harter. Expected values of normal order statistics. *Biometrika*, 48:151–165, 1961.
- [63] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, Cambridge, England, 2002.
- [64] Haikady N. Nagaraja Herbert A. David. *Order Statistics*. John Wiley & Sons, Hoboken, NJ, USA, 2003.
- [65] Bruce Hughes and Shmuel Weinberger. Surgery and stratified spaces. *Surveys on Surgery Theory*, pages 311–342, 2000.
- [66] Shalev Itzkovitz, Ron Milo, Nadav Kashtan, G. Ziv, and Uri Alon. Subgraphs in random networks. *Physical Review E*, 68:026127, 2003.
- [67] Matthew Kahle. Random geometric complexes. Submitted, 2009.
- [68] Mehmet Koyutürk, Ananth Grama, and Wojciech Szpankowski. Assessing significance of connectivity and conservation in protein interaction networks. *Journal of Computational Biology*, 14:747–764, 2007.
- [69] Martin Kulldorff. A spatial scan statistic. *Communications in Statistics: Theory and Methods*, 26:1481–1496, 1997.
- [70] Martin Kulldorff. Prospective time periodic geographical disease surveillance using a scan statistic. *Journal of the Royal Statistical Society, Series A*, 164:61–72, 2001.
- [71] Martin Kulldorff. Satscan user guide, 7.0 edition, August 2006.

- [72] Martin Kulldorff, Toshiro Tango, and Peter J. Park. Power comparisons for disease clustering tests. *Computational Statistics and Data Analysis*, 42:665–684, 2003.
- [73] Gilad Lerman and Teng Zhang. Probabilistic recovery of multiple subspaces in point clouds by geometric lp minimization, 2010.
- [74] John Mather. Notes on topological stability. Harvard University, 1970.
- [75] F. J. McErlean, David A. Bell, and Sally I. McClean. The use of simulated annealing for clustering data in databases. *Information Systems*, 15:233–245, 1990.
- [76] Carl McTague. Stratified morse theory. Unpublished expository essay written for Part III of the Cambridge Tripos, 2005.
- [77] Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H. Barr. Discrete differential-geometry operators for triangulated 2-manifod. *Visualization and Mathematics III*, pages 35–57, 2003.
- [78] Philippos Mordohai and Gérard Medioni. Dimensionality estimation, manifold learning and function approximation using tensor voting. *Journal of Machine Learning Research*, 11:411450, 2010.
- [79] Dmitriy Morozov. *Homological Illusions of Persistence and Stability*. PhD thesis, Duke University, 2008.
- [80] Jean-Marie Morvan. *Generalized Curvature*. Springer-Verlag, Berlin, Germany, 2008.
- [81] Stefanie Muff, Francesco Rao, and Amedeo Caffisch. Local modularity measure for network clusterizations. *Physical Review E*, 72:056107, 2005.
- [82] James R. Munkres. *Elements of Algebraic Topology*. Addison-Wesley, Redwood City, CA, USA, 1984.
- [83] Daniel Neill and Andrew Moore. Rapid detection of significant spatial clusters. In *Proceedings 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 256–265, 2004.
- [84] Daniel Neill, Andrew Moore, and Gregory Cooper. A bayesian scan statistic for spatial cluster detection. In *Proceedings of the National Syndromic Surveillance Conference*, 2005.

- [85] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133, 2004.
- [86] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74:036104, 2006.
- [87] M. E. J. Newman. Modularity and community structure in networks. In *Proceedings National Academy of Sciences*, volume 103, pages 8577–8582, 2006.
- [88] Partha Niyogi, Stephen Smale, and Shmuel Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete Computational Geometry*, 39:419–441, 2008.
- [89] Partha Niyogi, Stephen Smale, and Shmuel Weinberger. A topological view of unsupervised learning from noisy data. Manuscript, 2008.
- [90] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814–818, 2005.
- [91] Jagdish K. Patel and Campbell B. Read. *Handbook of the normal distribution*. CRC Press, Boca Raton, FL, USA, 1996.
- [92] Eric F. Pettersen, Thomas D. Goddard, Conrad C. Huang, Gregory S. Couch, Daniel M. Greenblatt, Elaine C. Meng, and Thomas E. Ferrin. Ucsf chimera - a visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, 25:1605–1612, 2004.
- [93] Markus J. Pflaum. *Analytic and Geometric Study of Stratified Spaces*. Springer-Verlag, Berlin, Germany, 2001.
- [94] David A. Plaisted. A heuristic algorithm for small separators in arbitrary graphs. *SIAM Journal on Computing*, 19:267 – 280, 1990.
- [95] Alex Pothen, Horst D. Simon, and Kan-Pu Liou. Partitioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications*, 11:430–452, 1990.
- [96] Carey E. Priebe, John M. Conroy, David J. Marchette, and Youngser Park. Scan statistics on enron graphs. *Computational and Mathematical Organization Theory*, 11:229–247, 2005.

- [97] Josep M. Pujol, Javier Béjar, and Jordi Delgado. Clustering algorithm for determining community structure in large networks. *Physical Review E*, 74:016107, 2006.
- [98] Jörg Reichardt and Stefan Bornholdt. Statistical mechanics of community detection. *Physical Review E*, 74:016110, 2006.
- [99] John A. Rice. *Mathematical Statistics and Data Analysis*. Duxbury Press, Belmont, CA, USA, 2003.
- [100] Sheldon M. Ross. *Stochastic Processes*. John Wiley & Sons, New York, NY, USA, 1995.
- [101] Colin Rourke and Brian Sanderson. Homology stratifications and intersection homology. *Geometry and Topology Monographs*, 2:455–472, 1999.
- [102] Michel F. Sanner, Arthur J. Olson, and Jean-Claude Spehner. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers*, 38:305–320, 1996.
- [103] Luis A. Santaló. *Integral Geometry and Geometric Probability*. Cambridge University Press, Cambridge, England, 2004. 1976 First Edition Addison-Wesley.
- [104] Benno Schwikowski, Peter Uetz, and Stanley Fields. A network of protein–protein interactions in yeast. *Nature Biotechnology*, 18:1257 – 1261, 2000.
- [105] Roded Sharan, Trey Ideker, Brian Kelley, Ron Shamir, and Richard M. Karp. Identification of protein complexes by comparative analysis of yeast and bacterial protein interaction data. *Journal of Computational Biology*, 12:835–846, 2005.
- [106] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 2000.
- [107] Ning-Zhong Shi and Jian Tao. *Statistical Hypothesis Testing: Theory and Methods*. World Scientific, Singapore, 2008.
- [108] William T. Tutte. *Graph Theory*. Cambridge University Press, Cambridge, England, 2001.
- [109] Stijn M. van Dongen. *Graph clustering by flow simulation*. PhD thesis, University of Utrecht, May 2000.
- [110] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1945 – 1959, 2005.

- [111] Jorma Virtamo. Queueing theory - Poisson process. Lecture Notes, 2010.
- [112] Yusu Wang, Pankaj K. Agarwal, Paul Brown, Herbert Edelsbrunner, and Johannes Rudolph. Coarse and reliable geometric alignment for protein docking. In *Proceedings of the Pacific Symposium on Biocomputing*, pages 66–77, 2005.
- [113] Shmuel Weinberger. *The topological classification of stratified spaces*. University of Chicago Press, Chicago, IL, USA, 1994.
- [114] Emo Welzl. Smallest enclosing disks (balls and ellipsoids). *Lecture Notes in Computer Science*, 555:359–370, 1991.
- [115] Dennis M. Wilkinson and Bernardo A. Huberman. A method for finding communities of related genes. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 101, pages 5241–5248, 2004.
- [116] Afra J. Zomorodian and Herbert Edelsbrunner. Fast software for box intersections. *International Journal of Computational Geometry and Applications*, 12:143–172, 2002.

Glossary

\mathcal{A} -small Given a topological space X and a collection \mathcal{A} of subsets of X whose interiors form an open cover of X , a singular simplex of X is said to be \mathcal{A} -small if its image set is entirely contained in a single element of \mathcal{A} [82]. 137

absolute Gaussian curvature Letting x be a point in the 2-manifold \mathbb{M} and $r > 0$, we define the absolute Gaussian curvature at x by taking the limit of a fraction of areas, $g(x) = \lim_{r \rightarrow 0} \frac{\text{Area}(N(A_r))}{\text{Area}(A_r)}$, where A_r is the neighborhood of points at distance at most r from x on \mathbb{M} , $N(A_r)$ is its area under the Gauss map. The total absolute Gaussian curvature is the integral of the local quantity, $G(\mathbb{M}) = \int_{x \in \mathbb{M}} g(x) dx$. 77, 183

absolute Gaussian curvature (PL case) See **Gaussian curvature (PL case)**. 80

alternative hypothesis See **hypothesis**. 17

bipartite A graph $G = (V, E)$ is bipartite if there is a partition of $V = X \cup Y$, with X and Y disjoint, such that each edge of G has one end in X and one in Y [108]. 15

birth (persistence module) Assume the persistence module \mathcal{F} is tame and so we have a finite ordered list of critical values $0 = c_0 < c_1 < \dots < c_m$. We choose regular values $\{a_i\}_{i=0}^m$ such that $c_{i-1} < a_{i-1} < c_i < a_i$ for all $1 \leq i \leq m$, and we adopt the shorthand notation $F_i = F_{a_i}$ and $f_i^j : F_i \rightarrow F_j$, for $0 \leq i \leq j \leq m$. A vector $v \in \mathcal{F}_i$ is said to be born at level i if $v \notin \text{im } f_{i-1}^i$, and such a vector dies at level j if $f_i^j(v) \in \text{im } f_{i-1}^j$ but $f_i^{j-1}(v) \notin \text{im } f_{i-1}^{j-1}$. Its birth time is c_i while its death time is c_j .

74, 94, 175, 180

birth (persistent homology) Given the lower star filtration and the sequence of homology groups connected by homomorphisms, letting γ be a class in $H_p(K_i)$, it is born at K_i if $\gamma \notin H_p^{i-1,i}$. If γ is born at K_i , then it dies entering K_j if it merges with an older class as we go from K_{j-1} to K_j , that is, $f_p^{i,j}(\gamma) \notin H_p^{i-1,j-1}$ but $f_p^{i,j}(\gamma) \in H_p^{i-1,j}$. If γ is born at K_i and dies entering K_j , then we call the difference in function value the persistence, $\text{pers}(\gamma) = a_j - a_i$. See **lower star filtration**. 51, 175, 180

bottleneck distance The bottleneck distance between any two persistence diagrams D and D' to is, $d_B(D, D') = \inf_{\Gamma: D \rightarrow D'} \sup_{u \in D} \|u - \Gamma(u)\|_\infty$, where Γ ranges over all bijections from D to D' . 95

cokernel See **kernel**. 97

cone The cone of a topological space \mathbb{X} is the quotient space $(\mathbb{X} \times I)/\mathbb{X} \times \{0\}$ of the product of \mathbb{X} with the unit interval $I = [0, 1]$ [63]. 100

critical point Let $f : \mathbb{M} \rightarrow \mathbb{R}$ be a smooth function on a d -manifold, a point $x \in \mathbb{M}$ is critical iff all its first-order partial derivatives vanish; otherwise it is regular. The image of a critical point is a critical value of f . All others are regular values of f . 72, 174, 181

critical region For each $u \in \mathbb{S}^2$, letting $h_u : |K| \rightarrow \mathbb{R}$ be the height function defined by $h_u(x) = \langle x, u \rangle$, the critical region of a vertex is the closure of the set of directions u along which it is critical. 80

critical value See **critical point**. 72

critical value (persistence module) See **regular value (persistence module)**. 94

critical vertex Given a generic function $f : |K| \rightarrow \mathbb{R}$ on the underlying space of a simplicial complex K , a vertex in K is critical if its lower link is not contractible; otherwise it is regular.. 50, 79, 181

death (persistence module) See [birth \(persistence module\)](#). 74, 94

death (persistent homology) See [birth \(persistent homology\)](#). 51

deformation retract Let X be a topological space and A a subspace of X . A continuous map $F : X \times [0, 1] \rightarrow X$ is a deformation retraction if for every $x \in X$ and $a \in A$, $F(x, 0) = x$, $F(x, 1) \in A$, and $F(a, 1) = a$. A deformation retraction is a homotopy between a retraction and the identity map on X . The subspace A is called a deformation retract of X . 137

Delaunay triangulation The dual of the Voronoi diagram of a set of points is the Delaunay triangulation. 123

distance function Given a topological space X embedded in some Euclidean space \mathbb{R}^n , we define d_X as the distance function which maps each point in the ambient space to the distance from its closest point in X . 98

elevation function For each $u \in \mathbb{S}^2$, suppose the height function h_u on the 2-manifold M is Morse, the points in the persistence diagram of h_u correspond to pairs of critical points. The elevation at the points x and y of such a pair is set to the absolute height difference in the direction, $E(x) = E(y) = |h_u(x) - h_u(y)|$. There is a unique value at every point in M . This is the elevation function, $E : M \rightarrow \mathbb{R}$. 76

excision Given topological spaces X , A and U such that $U \subseteq A \subseteq X$, the inclusion map of pairs $(X - U, A - U) \rightarrow (X, A)$ is called an excision if it induces a homology isomorphism [60]. 136

extended real plane The extended real plane is defined as $\bar{\mathbb{R}}^2 = (\mathbb{R} \cup \{\pm\infty\})^2$. 52, 94

feature size If the persistence module $\mathcal{F}_{\mathbb{R}_{\geq 0}}$ is tame, then it has a smallest non-zero critical value $\rho(\mathcal{F})$, we call this number the feature size of the persistence module. 94

Gauss map The Gauss map on a 2-manifold, $N : M \rightarrow \mathbb{S}^2$, maps a point $x \in M$ to the outer unit normal at x . 77

Gaussian curvature (PL case) The Gaussian curvature of a vertex in a triangulated surface is the area of its critical region weighted by the winding number. Its absolute Gaussian curvature is defined as the area weighted by the absolute winding number. The total absolute Gaussian curvature is then the sum over all vertices. 80, 173, 183

graph A graph $G = (V, E)$ is defined by a set V of elements called *vertices*, a set E of elements called *edges*, and a relation of *incidence*, which associates with each edge either one or two vertices called its *ends* [108]. 1, 10

half-normal The half-normal distribution is the probability distribution of the absolute value of a random variable that is normally distributed with expected value 0 and variance σ^2 . 53

height function Let \mathbb{M} be a smoothly embedded 2-manifold in \mathbb{R}^3 . Given a direction $u \in \mathbb{S}^2$, the height function in this direction, $h_u : \mathbb{M} \rightarrow \mathbb{R}$, is defined by mapping each point x to $h_u(x) = \langle x, u \rangle$. 75

Hessian The Hessian of a function f at a point x is the matrix of second partial derivatives at the point. 72

homology group The p -th homology group is the p -th cycle group modulo the p -th boundary group. 51

homotopy A homotopy between two continuous functions f and g from a topological space \mathbb{X} to a topological space \mathbb{Y} is defined to be a continuous function $H : \mathbb{X} \times [0, 1] \rightarrow \mathbb{Y}$ from the product of the space \mathbb{X} with the unit interval $[0, 1]$ to \mathbb{Y} such that, if $x \in \mathbb{X}$ then $H(x, 0) = f(x)$ and $H(x, 1) = g(x)$. 59

homotopy equivalent Two spaces X and Y are said to be homotopy equivalent, or to have the same homotopy type, if there are maps $f : X \rightarrow Y$ and $g : Y \rightarrow X$ such that $g \circ f$ is homotopic to the identity map id_X on X , and $f \circ g$ is homotopic to the identity map id_Y on Y . 50

hypothesis A hypothesis is a statement about a population parameter. The two complementary hypotheses in hypothesis testing are called the null hypothesis and the alternative hypothesis, denoted by H_0 and H_1 , respectively. If θ denotes a population parameter, the general format of the null and alternative hypothesis is $H_0 : \theta \in \Theta_0$, $H_1 : \theta \in \Theta_0^c$, where Θ_0 is some subset of the parameter space Θ and Θ_0^c is its complement. A hypothesis testing is a rule that specifies: (i) For which sample values the decision is made to accept H_0 as true; (ii) For which sample values H_0 is rejected and H_1 is accepted as true [24]. 17, 173, 177, 179

hypothesis testing See hypothesis. 18

index According to Morse Lemma, the number of minus signs in the quadratic polynomial is the index of the critical point [50]. 72

individually most powerful The parameter space Θ_0^c is partitioned into a countable number of subsets $\{A_j\}$. Likewise using the same index, the critical region R is partitioned into subsets $\{R_j\}$. Let R' denote an alternative critical region with corresponding disjoint subsets, $\{R'_j\}$. A test is individually most powerful with respect to a partition $\{A_j\}$ of the parameter space Θ_0^c , and a partition $\{R_j\}$ of the critical region R , if for each A_k there are no sets R' and $\{R'_j\}$ such that: assuming a significance level α , 1) $\beta(\theta) = \beta'(\theta)$ if $\theta \notin A_k$; 2) $\beta(\theta) < \beta'(\theta)$ if $\theta \in A_k$. 20

kernel If $f : U \rightarrow V$ is a linear transformation between vector spaces, the kernel, image and cokernel are defined as, $\ker f = \{u \in U | f(u) = 0 \in V\}$, $\text{im } f = \{v \in V | \text{there exists } u \in U \text{ with } f(u) = v\}$, $\text{cok } f = V/\text{im } f$ [50]. 97, 174

likelihood function A likelihood function is a function of the parameters of a statistical model that allows estimation of unknown parameters based on known outcomes. Formally, let $f(x|\theta)$ denote the probability density function of the random variable X . That is, over any range R , $\Pr(X \in R) = \int_{x \in R} f(x|\theta)dx$, based on a known pa-

parameter θ . Then, the function of θ defined by $L(\theta|x) = f(x|\theta)$ is called the likelihood function [24]. 16

likelihood ratio test See [likelihood ratio test statistic](#). 18

likelihood ratio test statistic The likelihood ratio test statistic for testing $H_0 : \theta \in \Theta_0$ versus $H_1 : \theta \in \Theta_0^c$ is defined as follows, which equals the ratio of the maximum likelihood values, $\lambda(x) = \frac{\sup_{\Theta_0} L(\theta|x)}{\sup_{\Theta} L(\theta|x)} = \frac{L(\hat{\theta}_0|x)}{L(\hat{\theta}|x)}$. A likelihood ratio test (LRT) is any test that has a critical region of the form $\{x : \lambda(x) \leq c\}$ for some $0 \leq c \leq 1$ [24]. 18, 178

link The link of a vertex consists of all faces of simplices in the star that do not belong to the star. 50, 78

local feature size The local feature size of a point $a \in \mathbb{X}$ is the distance of a to the medial axis of \mathbb{X} . 113

local homology The local homology groups of a space \mathbb{X} at a point $x \in \mathbb{X}$ are the groups $H_i(\mathbb{X}, \mathbb{X} - x)$ in each homological dimension i [82]. 100

lower link Given a piecewise-linear function $f : |K| \rightarrow \mathbb{R}$, the lower link of a vertex is the subset of simplices in the link with smaller function values than the vertex. 50, 78

lower star Given a piecewise-linear function $f : |K| \rightarrow \mathbb{R}$, the lower star of a vertex u is the subset of simplices in the star of u for which u is the vertex with the maximum function value. 50, 78

lower star filtration Let K be a triangulation of a topological space. Given a generic function $f : |K| \rightarrow \mathbb{R}$, the lower star filtration is the sequence $\emptyset = K_0 \subseteq K_1 \subseteq \dots \subseteq K_n = K$, where K_i is the union of the lower stars of the first i vertices in the ordering by f . In other words, if $a_1 < a_2 < \dots < a_n$ are the function values of the vertices in K and $a_0 = -\infty$, then $K_i = K(a_i) = \bigcup_{u \in V, f(u) \leq a_i} \text{St}_- u$ for each i . 50,

maximum likelihood estimator For sample point x , let $\hat{\theta}(x)$ be a parameter value at which $L(\theta|x)$ obtains its maximum as a function of θ with x fixed. A maximum likelihood estimator (MLE) of the parameter θ based on a sample X is $\hat{\theta}(X)$ [24]. 17

medial axis The medial axis \mathcal{M} of an embedded space \mathbb{X} is the subset of the ambient space consisting of all points which have at least two nearest neighbors on \mathbb{X} . 113

minimal carrier (chain) See **minimal carrier (simplex)**. 134

minimal carrier (simplex) The minimal carrier of a singular simplex is its image. The minimal carrier of a singular chain is the union of the minimal carriers of its simplices [82]. 134, 179

Morse function $f : \mathbb{M} \rightarrow \mathbb{R}$ is a Morse function if all its critical points are non-degenerate and its values at the critical points are distinct. 73

Morse Lemma Let u be a non-degenerate critical point of $f : \mathbb{M} \rightarrow \mathbb{R}$ on a smooth d -manifold. There are local coordinates with $u = (0, 0, \dots, 0)$ such that $f(x) = f(u) - x_1^2 - \dots - x_q^2 + x_{q+1}^2 + \dots + x_d^2$, for every point $x = (x_1, x_2, \dots, x_d)$ in a small neighborhood of u [50]. 177

nerve The nerve of a finite collection of sets consists of all non-empty subcollections whose sets have a non-empty common intersection. 122

non-degenerate critical point A critical point x is non-degenerate if the Hessian at the point is non-singular. 72

normal The normal distribution is a continuous probability distribution whose probability density function is, $f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$, where μ and σ are the mean and variance.

null hypothesis See **hypothesis**. 17

persistence (persistence module) See [persistence diagram \(persistence module\)](#). 74

persistence (persistent homology) See [birth \(persistent homology\)](#). 51

persistence diagram (persistence module) Given a persistence module \mathcal{F} , let $P^{i,j}$ be the vector space of vectors that are born at level i (with critical value c_i) and then subsequently die at level j (with critical value c_j), and $\beta^{i,j}$ denotes its rank. The persistence of these vectors is $c_j - c_i$. The persistence diagram of \mathcal{F} , $\text{Dgm}(\mathcal{F})$, contains a multiset of points in the extended real plane. It contains $\beta^{i,j}$ copies of the points (c_i, c_j) , as well as infinitely many copies of each point along the major diagonal $y = x$. See [persistence module](#) and [birth \(persistence module\)](#). 94, 180

persistence diagram (persistent homology) The p -th persistence diagram of a function f , $\text{Dgm}_p(f)$, contains a multiset of points in the extended real plane. It contains β_p copies of the points that represent the birth and death of p -dimensional homology classes, as well as infinitely many copies of each point along the major diagonal $y = x$. 52

persistence module Let A be some subset of \mathbb{R} . A persistence module \mathcal{F}_A is a family $\{F_\alpha\}_{\alpha \in A}$ of $\mathbb{Z}/2\mathbb{Z}$ -vector spaces, together with a family $\{f_\alpha^\beta : F_\alpha \rightarrow F_\beta\}_{\alpha \leq \beta \in A}$ of linear maps such that $\alpha \leq \beta \leq \gamma$ implies $f_\alpha^\gamma = f_\beta^\gamma \circ f_\alpha^\beta$. 93, 180

persistent homology group Given the lower star filtration, for each $i \leq j$, the inclusion map $K_i \rightarrow K_j$ induces homomorphisms between homology groups, $f_p^{i,j} : H_p(K_i) \rightarrow H_p(K_j)$, for dimension p . This gives a sequence of homology groups connected by homomorphisms, $0 = H_p(K_0) \rightarrow H_p(K_1) \rightarrow \dots \rightarrow H_p(K_n) = H_p(K)$, the p -th persistent homology groups are the images of the homomorphisms induced by inclusion, $H_p^{i,j} = \text{im } f_p^{i,j}$ for $0 \leq i, j \leq n$. The corresponding p -th persistent Betti numbers β_p are the ranks of these groups. 51

point cloud A point cloud is a set of vertices in \mathbb{R}^k . 1

Poisson process A Poisson process is a stochastic process in which events occur continuously and independently of one another. Formally, a counting process $\{N(t), t \geq 0\}$ is said to be a Poisson process having rate λ , $\lambda > 0$, if: 1) $N(0) = 0$; 2) The process has independent increments; 3) The number of events in any interval of length t is Poisson distributed with mean λt [100]. 10

power The power of a test is the probability that the test will not make Type II error. 19

power function The power function of a hypothesis test with rejection region R is the function of θ defined by $\beta(\theta) = P_\theta(X \in R)$ [24]. For $\theta \in \Theta_0$, it equals the probability of a Type I Error. For $\theta \in \Theta_0^c$, it equals one minus the probability of a Type II Error. 19

reach The reach of a topological space \mathbb{X} is the smallest distance between \mathbb{X} and its medial axis. 113

reduced Betti number For a non-negative integer p , the p -th reduced Betti number is the rank of the p -th reduced homology group of \mathbb{X} , $\tilde{H}_p(\mathbb{X})$ [82]. 50

regular point See [critical point](#). 72

regular value See [critical point](#). 72

regular value (persistence module) A real number α is said to be a regular value of the persistence module \mathcal{F} if there exists some $\epsilon > 0$ such that, for all $\delta < \epsilon$, the maps $f_{\alpha-\delta}^{\alpha+\delta}$ are all isomorphisms. Otherwise we say that α is a critical value of the persistence module; if $A = \mathbb{R}_{\geq 0}$, then $\alpha = 0$ will always be considered to be a critical value. 93, 174

regular vertex See [critical vertex](#). 50, 79

rejection region The subset of the sample space for which H_0 is rejected is called the rejection region or critical region. 18

significance level For $0 \leq \alpha \leq 1$, a test with power function $\beta(\theta)$ is a level α test if $\sup_{\theta \in \Theta_0} \beta(\theta) \leq \alpha$ [24]. α is called the significance level. 19

simplicial complex A simplicial complex K in \mathbb{R}^n is a collection of simplices in \mathbb{R}^n such that every face of a simplex of K is in K , and the intersection of any two simplices of K is a face of each of them [82]. 49, 78

singular chain group The free abelian group generated by the singular p -simplices of X is called the singular chain group of X in dimension p . 134

singular simplex A singular p -simplex of a topological space X is a continuous map $\delta : \Delta_p \rightarrow X$, where Δ_p is the standard p -simplex [82]. 134

standard simplex The standard p -simplex is the subset of \mathbb{R}^{p+1} , given by $\Delta_p = \{(t_0, \dots, t_p) \in \mathbb{R}^{p+1} \mid \sum_{i=0}^p t_i = 1, \forall i, t_i \geq 0\}$ [82]. 134

star The star of a vertex is the set of simplices that contain it. 50, 78

stratification A d -dimensional stratification of a topological space \mathbb{X} is a decreasing sequence of closed subspaces $\mathbb{X} = \mathbb{X}_d \supseteq \mathbb{X}_{d-1} \supseteq \dots \supseteq \mathbb{X}_0 \supseteq \mathbb{X}_{-1} = \emptyset$, such that for each i , the i -dimensional stratum $\mathbb{S}_i = \mathbb{X}_i - \mathbb{X}_{i-1}$ is a (possibly empty) i -manifold. 99

Stratified Morse function f is a Stratified Morse function iff: 1) f is a Morse function when restricted to each manifold piece; 2) All critical values of f are distinct; 3) The differential of f at a critical point $x \in S_i$ does not annihilate any generalized tangent space to x other than $T_x S_i$. 102

strongly interleaved Two persistence modules \mathcal{F} and \mathcal{G} are said to be strongly ϵ -interleaved if, for some positive ϵ , there exist two families $\{\xi_\alpha : F_\alpha \rightarrow G_{\alpha+\epsilon}\}_\alpha$ and $\{\psi_\alpha : G_\alpha \rightarrow F_{\alpha+\epsilon}\}$ of linear maps which commute with the module maps $\{f_\alpha^\beta\}$ and $\{g_\alpha^\beta\}$ in the appropriate manner. More precisely, we require, for all $\alpha \leq \beta$, $f_{\alpha-\epsilon}^{\beta+\epsilon} = \psi_\beta \circ g_\alpha^\beta \circ \xi_{\alpha-\epsilon}$ and $\psi_\beta \circ g_\alpha^\beta = f_{\alpha+\epsilon}^{\beta+\epsilon} \circ \psi_\alpha$, as well as the two other equations obtained by exchanging

the roles of f and g and ξ and ψ [26]. 95

sublevel set Let $f : \mathbb{M} \rightarrow \mathbb{R}$ be a smooth function on a d -manifold. Given $a \in \mathbb{R}$, the sublevel set consists of all point with value at most a , $\mathbb{M}_a = f^{-1}(-\infty, a]$. The superlevel set is $\mathbb{M}^a = f^{-1}[a, \infty)$. 73, 183

superlevel set See **sublevel set**. 74

tame A persistence module \mathcal{F} is tame if it has a finite number of critical values and if all the vector spaces F_α are of finite rank. 94

total absolute Gaussian curvature See **absolute Gaussian curvature**. 77

total absolute Gaussian curvature (PL case) See **Gaussian curvature (PL case)**. 80

total persistence The total persistence of a persistence diagram is the sum of the persistences of its points. 52

triangulation A triangulation of a topological space \mathbb{X} is a simplicial complex K together with a homeomorphism between \mathbb{X} and $|K|$. 1, 49, 78

Type I Error If $\theta \in \Theta_0$ but the test incorrectly decides to reject H_0 , it makes the Type I Error, the probability of making Type I error is denoted as $P_\theta(X \in R)$, which is commonly known as the false positive rate. 18

Type II Error If $\theta \in \Theta_0^c$ but the test incorrectly decides to accept H_0 , it makes the Type II Error, the probability of making Type II error is denoted as $P_\theta(X \in R^c) = 1 - P_\theta(X \in R)$, which is commonly known as the false negative rate. 19

uniformly most powerful Let \mathcal{C} be a class of tests for testing $H_0 : \theta \in \Theta_0$ versus $H_1 : \theta \in \Theta_0^c$. A test in class \mathcal{C} with power function $\beta(\theta)$, is a uniformly most powerful (UMP) class \mathcal{C} test if $\beta(\theta) \geq \beta'(\theta)$ for every $\theta \in \Theta_0^c$ and every $\beta'(\theta)$ that is a power function of a test in class \mathcal{C} [24]. Simply put, a test is UMP if it has a smaller Type II error than all other tests in the same class. 19

Voronoi cell Given a set of points in \mathbb{R}^n , the Voronoi cell of a point $u \in \mathbb{R}^n$ is the set of points for which u is the closest. 123

Voronoi diagram The Voronoi diagram of a set of points is the collection of Voronoi cells of its points. 123

Whitney stratification A stratification is called a Whitney stratification if for every pair of strata pieces S_i and S_j with $S_i \subset \text{cl } S_j$, the Whitney conditions A and B hold. 101

winding number Given a closed curve on \mathbb{S}^2 , the winding number of a direction $u \in \mathbb{S}^2$ not on the curve is the number of times the curve goes around the directed line defined by u . Viewed along u , we count a counterclockwise turn as $+1$ and a clockwise turn as -1 . Taking the sum we get the winding number. 80

Biography

Bei Wang was born in Chengdu, China on July 15, 1981. She received the Bachelor of Science degree summa cum laude in Computer Science, Bachelor of Science degree in Mathematics, with a minor in Psychology, from University of Bridgeport in 2003. She had one year of graduate study in Computer Science at Stony Brook University before transferring to Duke University in 2004 to continue her pursuit of PhD degree in Computer Science.

At University of Bridgeport, Bei received Phi Kappa Phi Award of Excellence, Sigma Xi Grant-in-Aid of Research and Upsilon Pi Epsilon Microsoft Scholarship Award. At Duke she was awarded the best graduate teaching assistant. Bei's research interests lie in computational geometry and topology, computational biology and bioinformatics.

1. Bei Wang, Herbert Edelsbrunner and Dmitriy Morozov. Computing elevation maxima by searching the Gauss sphere. *Lecture Notes in Computer Science*, 5526, pages 281-292, 2009.

2. Mats Ensterö, Örjan Åkerborg, Daniel Lundin, Bei Wang, Terrence Furey, Marie Öhman and Jens Lagergren. A computational screen for site selective A-to-I editing detects novel sites in neuron specific Hu proteins. *BMC Bioinformatics*, 11(6), 2009.

3. Bei Wang, Jeff M. Phillips, Robert Schrieber, Dennis Wilkinson, Nina Mishra and Robert Tarjan. Spatial Scan Statistics for Graph Clustering. In *Proceedings of the 8th SIAM International Conference on Data Mining*, 2008.

4. Sudheer Sahu, Bei Wang and John Reif. A Framework for Modeling DNA Based

Molecular Systems. Lecture Notes in Computer Science, 4287, pages 250-265, 2006.

5. Bei Wang, Dimitris Papamichail, Steffen Mueller and Steven Skiena. Two Proteins for the Price of One: The Design of Maximally Compressed Coding Sequences. Lecture Notes in Computer Science, 3892, pages 387-398, 2006.