

DISSERTATION

TOPOLOGICAL TECHNIQUES FOR CHARACTERIZATION
OF PATTERNS IN DIFFERENTIAL EQUATIONS

Submitted by

Rachel A. Neville

Department of Mathematics

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2017

Doctoral Committee:

Advisor: Patrick Shipman

Henry Adams

Amber Krummel

Clayton Shonkwiler

Copyright by Rachel A. Neville 2017

All Rights reserved

ABSTRACT

TOPOLOGICAL TECHNIQUES FOR CHARACTERIZATION OF PATTERNS IN DIFFERENTIAL EQUATIONS

Complex data can be challenging to untangle. Recent advances in computing capabilities has allowed for practical application of tools from algebraic topology, which have proven to be useful for qualitative and quantitative analysis of complex data. The primary tool in computational topology is persistent homology. It provides a valuable lens through which to study and characterize complex data arising as orbits of dynamical systems and solutions of PDEs. In some cases, this includes leveraging tools from machine learning to classify data based on topological characteristics. We see a unique pattern arising in the persistence diagram of a class of one-dimensional discrete dynamical systems—even in chaotic parameter regimes, and connect this to the dynamics of the system in Chapter 2. Geometric pattern structure tell us something about the parameters driving the dynamics in the system as is the case for anisotropic Kuramoto-Sivashinsky equation which displays chaotic bubbling. We will see this in Chapters 3 and 4. Defects in pattern-forming systems be detected and tracked and studied to characterize the degree of order in near-hexagonal nanodot structures formed by ion bombardment, which will be developed in Chapter 5.

ACKNOWLEDGEMENTS

I begin with deep gratitude towards God who, most importantly, has chosen me as his own and rescued me from my sin, and who has graciously placed me at CSU for this season and enabled me to complete this thesis. Without him, there is nothing good in me. I feel very fortunate to have a chance to study mathematics, through which I have been given a glimpse of the infinite beauty and structure of the created world that few get to glimpse.

I am indebted to my advisor, Patrick Shipman, for his mentorship and friendship. He has believed in me, patiently encouraged me and gently challenged my self imposed limitations. My time working with him has been invaluable and has shaped me to be the mathematician that I am today. Patrick sees utter joy in learning, has an eye for the beautiful, overarching themes of mathematics and draws his students into his fascination.

I am thankful for the support and input of my committee. I count myself fortunate to be in the mathematics department at CSU. The spirit of collaboration due to the professors and graduate students makes this a joyful department to be in. A number of professors and peers have offered valuable input and encouragement over the years. I have enjoyed the Topological Data Analysis Group, Pattern Analysis Lab, and Ion Bombardment Group where I've had the chance to learn from lots of different people and delve into interesting corners of math.

I have enjoyed the valuable conversations with fellow graduate students on math and life and everything in between. I have been challenged to become a better scientist, learner, teacher and friend through these conversations. Thank you in particular to those who have offered their day in and day out support, you have been good friends to me. In particular, Vance Blankers for his enduring optimism, encouragement, and ever-willingness to challenge my thinking.

I am thankful for the incredible love and support of my friends and family, without which I would not be able to push through the hard and sometimes defeating days of graduate school. I appreciate my parents, who have who taught me to do my best and who have celebrated every little victory along the way.

Last, I am so grateful for my husband, Stephen. The way you have loved me, supported me, and believed in me through graduate school is a precious gift. Thank you for your patience grace towards me and for continuing to call me to the deeper and truer.

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGEMENTS	iii
LIST OF TABLES	vii
LIST OF FIGURES	viii
Chapter 1 Persistent Homology	1
1.1 Introduction	1
1.2 Homology	3
1.2.1 Homology of Simplicial Complexes	4
1.3 Persistent Homology	7
1.4 Building Simplicial Complexes	9
1.5 Discrete Morse Theory	11
1.5.1 Cubical Homology	13
1.5.2 Relative Homology	16
1.5.3 Persistent Homology of a Function	17
1.6 Persistence Images as a Classification Tool	19
Chapter 2 Persistent Homology and Unimodal Maps	23
2.1 Discrete Dynamical Systems	25
2.1.1 The Logistic Map	27
2.1.2 Pattern in the Persistence Diagram	30
2.1.3 The Tent Map	32
2.2 Persistence of Unimodal Maps	37
2.2.1 Examples of Unimodal Maps	40
2.3 Poincaré Maps of Higher Dimensional Systems	41
2.3.1 Noisy Maps	44
Chapter 3 Parameter Classification of Pattern-Forming Systems	48
3.1 Introduction	48
3.2 Pattern Formation	49
3.3 Cubical complexes and PDEs	51
3.4 Anisotropic Kuramoto–Sivashinsky Equation	53
3.4.1 Reduced Resolution and Variance Classification	54
3.4.2 Anisotropy Classification	56
3.4.3 Persistence of the aKS Equation	57
3.5 Application to Patterns Produced by Ion Bombardment	59
3.5.1 Varying Several Parameters	61
3.5.2 Varying a Linear Parameter	67
3.5.3 Varying a Nonlinear Parameter	73

Chapter 4	Characterization of Surface Properties	81
4.1	Introduction	81
4.2	Snow Roughness	81
4.3	Pyramidal Surface Structures	86
Chapter 5	Topological Measure of Order	91
5.1	Lattice Patterns	91
5.2	Measures of Order	92
5.2.1	Spectral, Geometric-Combinatorial, and Topological Tools for Describ- ing Lattice Patterns	93
	The Fourier Width at Half Maximum	93
	Autocorrelation Function	94
	Delaunay Triangulation	95
	Persistent Homology	95
5.3	Characterizations of Order	96
5.3.1	Bravais Lattices	97
	Measures of hexagonal order using the Delaunay triangulation	99
	The PH Measures of Order:	100
5.3.2	Variations on Filtrations	103
5.3.3	Bravais lattices perturbed at one point	104
5.3.4	Randomly perturbed Bravais lattices	105
5.4	Application to Patterns Produced by Ion Bombardment	108
5.4.1	Templated Surfaces	109
5.5	Conclusion	111
References	112

LIST OF TABLES

2.1	2 ⁿ -cycle bifurcations of the logistic map	28
2.2	Difference between actual and estimated parameter, noise added	45
2.3	Standard deviation between actual and estimated parameter, noise added	46
2.4	Standard deviation between actual and estimated parameter, noisy parameter	46
2.5	Standard deviation between actual and estimated parameter, noisy parameter	46
2.6	Difference between actual and estimated parameter, both types of noise	46
2.7	Standard deviation between actual and estimated parameter, both types of noise	47
3.1	Initial classification of solutions of the aKS equation with non-topological methods	56
3.2	Classification accuracies at different times of the aKS data	58
3.3	Classification accuracies of the aKS data for different PI resolutions	59
3.4	Classification accuracies of the aKS data for different PI variances	59
3.5	BS equation parameter set 1	62
3.6	Classification results of Ion Set 1	63
3.7	2D Rips PD feature check	65
3.8	Summary of classification accuracies as <i>b</i> varies	70
3.9	Classification of PIs computed on normalized surfaces, varying <i>b</i>	70
3.10	Classification of PIs for several time steps, varying <i>b</i>	71
3.11	Summary of classification accuracies as the nonlinear parameter <i>v</i> is varied.	75
3.12	Variances of solutions at t=25	77
3.13	Accuracies on normalized ion data, varying <i>v</i>	77
3.14	Classification results for multiple time steps, varying <i>v</i>	80
5.1	PH statistics for VR and Čech filtrations	104

LIST OF FIGURES

1.1	Examples of similar patterns in nature	1
1.2	Examples of similar patterns from different models	2
1.3	Example simplicial complex	4
1.4	Effect of boundary operator	7
1.5	A filtration of VR complexes and resulting barcode	9
1.6	Comparison of the Čech, VR and alpha complexes.	10
1.7	Three simplicial complexes with different ϵ	11
1.8	Filtration of cubical complexes.	14
1.9	A motivating example for 4-8 connectivity	15
1.10	Persistence of the sublevel sets of a one-dimensional function	18
1.11	Connection between Čech complexes and functional persistence	19
1.12	Persistence image pipeline	20
2.1	Cobweb diagram	26
2.2	Logistic map $r=3.52$	27
2.3	Logistic map $r=3.8$	27
2.4	Bifurcation diagram for the logistic map.	29
2.5	PD of randomly drawn points	31
2.6	PD of logistic map, periodic regime	31
2.7	PD of logistic map, chaotic regime	31
2.8	Updating function and intervals of the tent map	33
2.9	Iterations of the tent map	33
2.10	Slope of the PD for the tent map	34
2.11	Markov graph for the tent map	34
2.12	Markov graph for a unimodal map	38
2.13	A general unimodal map with the intervals marked.	40
2.14	Quadratic map and PD	41
2.15	Example of a trajectory of the Lorenz attractor.	42
2.16	Persistence of each dimension of the Lorenz attractor	42
2.17	Example of a trajectory of the Rössler attractor.	43
2.18	Persistence of each dimension of the Rössler attractor	43
2.19	Example of PD with fit, noise added in three ways	45
3.1	Examples of pattern formation	48
3.2	Cautionary example for cubical complexes of discretization	53
3.3	Solutions of the aKS equation	54
3.4	aKS variance histograms	56
3.5	Histogram of the angle of gradient vector of aKS classes	57
3.6	Soft mode influence	61
3.7	Examples of ion bombarded surfaces from data set 1	62
3.8	PDs from a 2D VR filtration on ion data set 1	65
3.9	PDs from a 3D VR filtration on ion data set 1	66

3.10	PDs from a Morse filtration on ion data set 1	67
3.11	Examples of solutions to the BS equations varying linear parameter b	68
3.12	PDs from a Morse filtration on ion data varying b	72
3.13	Examples of solutions to the BS equations varying nonlinear parameter ν	74
3.14	PDs from aMorse filtration as ν varies	75
3.15	PDs from a 3D VR filtration as ν varies	76
3.16	PDs from a 2D VR filtration as ν varies	76
4.1	PH of entire snow surface	81
4.2	PH sums of disjoint patches for Niwot data	84
4.3	Example of data, PH measures and persistence diagrams	84
4.4	Sliding window PH sums for Niwot data	85
4.5	Several methods of summarizing PH diagrams	85
4.6	Simulations of pyramidal data	87
4.7	Pyramid and inverted pyramid comparison	88
4.8	H_0 measure for pyramidal data evolving over time.	90
5.1	DT and Voronoi cells of defect filled hexagonal lattice	92
5.2	Fourier transform and radial average of an example of ion data	93
5.3	Several filtrations and the resulting persistence diagram of ion data	96
5.4	Parameterization of Bravais lattices and resulting PH	98
5.5	Delaunay triangulation and Voronoi cells of ion data	100
5.6	DT scores of parameterized Bravais lattices	102
5.7	PH scores of parameterized Bravais lattices, VR filtration	103
5.8	Measures of order for a perfect lattice with perturbed center	105
5.9	Normalized measures of order for noisy lattice	106
5.10	Examples of simulations with templated initial conditions	109
5.11	PH statistics for templated simulations	110

Chapter 1

Persistent Homology

1.1 Introduction

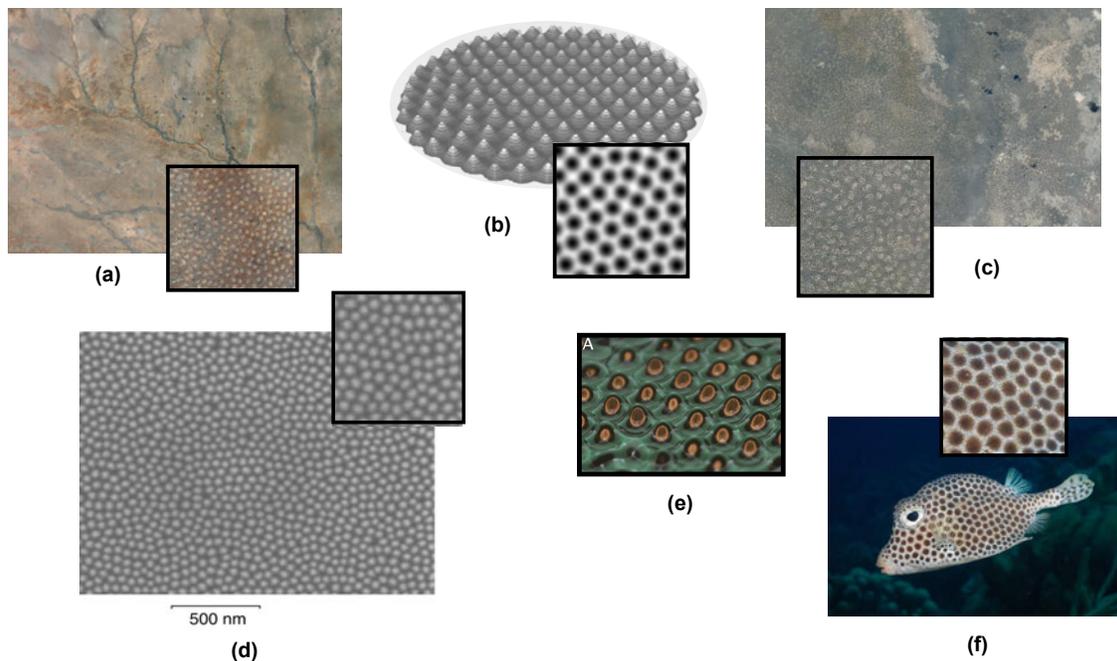


Figure 1.1: Examples of visually similar patterns appearing in different contexts, driven by different mechanisms and at vastly different scales. (a) Mineral pattern formation in soil, Somalia [58], (b) X-ray of ferrofluids under a magnetic field [49], (c) vegetation pattern in drylands, Ethiopia [58], (d) nanodots on GaSb surface that has been bombarded with ions [35], (e) Faraday waves, silicone oil droplets in water [10], (f) spotted trunkfish [51].

Much of applied mathematics is concerned with discovering patterns and structure from the world around us. As Robert Ghrist puts it “applied mathematics concerns the incarnation of mathematical objects and structures” [47]. Figure 1.1 shows visually similar patterns arising in vastly different settings, caused by vastly different mechanisms. In attempting to model such phenomena, many different models give rise to visually similar patterns, but mathematically incorporate very different mechanisms. See Figure 1.2, for example. How then might one com-

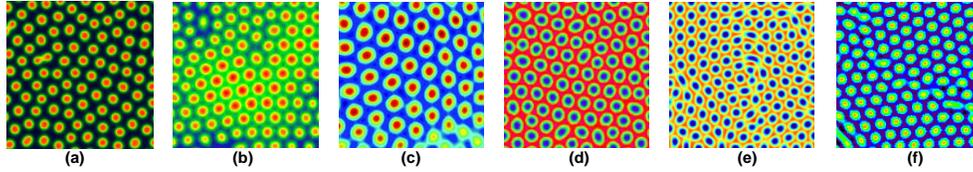


Figure 1.2: Visually similar patterns arising from models which incorporate different physical mechanisms. (a) Gray-Scott reaction-diffusion model [1], (b) Bradley-Shipman equations of ion bombardment, couples surface topography and sputter yield [7], (c) Benard Maragoni convection [120], (d) Swift-Hohenberg model [42], (e) Brusselator (modified reaction-diffusion model) [24], (f) modified Swift-Hohenberg equation [121]

pare visually similar patterns and models to better understand parameters and mechanisms that give rise to these patterns?

While there are many approached to this question, we will approach from the perspective of algebraic topology. Algebraic topology is a classical branch of mathematics that uses tools from abstract algebra to compare and characterize geometric objects. Computational implementations of the tools of algebraic topology have only recently become feasible due to massive increases in computing capabilities. The tools of computational topology allow for qualitative and quantitative analysis of complex data sets [11]. One of the primary tools is persistent homology (PH). This will be the main topological tool that we leverage in this thesis. PH has successfully been used to characterize complex structure in a number of data-driven applications such the neural code [21], biological swarms [113], and periodic signals [92], among many others. We will use PH as a way to exploit topological structure in complex data that arises as orbits of dynamical systems or solutions to partial differential equations (PDEs).

We start with a brief introduction to homology, and building to persistent homology, first on point cloud data and then in a functional setting. In chapters 3, 4 and 5, we will see each perspective lends itself naturally to different applications.

1.2 Homology

To leverage abstract algebra to compare and classify geometric spaces, we use algebraic invariants such as the fundamental group, which qualitatively describes our space.¹ We can compute the fundamental group of our space. The fundamental group is the number of classes of loops with a fixed based point that can be continuously deformed into each other along the surface of our object. This provides information about the number of holes in our space, which is topologically invariant. The natural higher-dimensional analog of the fundamental group are homotopy groups. If we wish to compare two different spaces, comparing their associated fundamental groups.

However, higher dimensional homotopy groups are very difficult to compute in general in part because these holes can interact with each other in complex ways. Homotopy groups are not directly computable from a cell structure [54], which is one way to handle discretized data.

Homology serves a similar purpose. Although not quite as intuitive, it is much more computable. Homology groups are finitely-generated abelian groups. We will be concerned with cycles, rather than the loops of homotopy groups, which are indeed loops, without a chosen base point. The dimension of the homology group will indicate the number of connected components, holes and higher dimensional voids of our object.

In this setting, we are after a way to characterize connected components, holes and higher dimensional voids of point cloud data, but to make any sense of this for a discrete sampling of points, we will need to take our discrete space and give it some additional structure. Instead of looking at deforming paths into each other, we will look at interesting boundary cycles of the structure we put on the data.

¹In broad strokes, we use homotopy to motivate homology. For a thorough introduction to homotopy, see Chapter 1 of Hatcher. [54]

1.2.1 Homology of Simplicial Complexes

We will assume some basic notions of algebra, but will take a moment to establish a few of the more technical building blocks of simplicial complexes. A $(k+1)$ -tuple of points (x_0, x_1, \dots, x_k) in \mathbb{R}^n is *affinely independent* if the set of vectors given by $\{x_0 - x_j | 1 \leq j \leq k\}$ is linearly independent. A p -simplex σ is the *convex hull* of $p+1$ affinely independent points and is denoted $\sigma = \text{conv}\{x_0, \dots, x_p\}$. The convex hull is the solid polyhedron determined by the $p+1$ vertices. For example, a 0-simplex is a vertex, a 1-simplex is an edge, a 2-simplex is a triangle and a 3-simplex is a tetrahedron.

A *face* of a simplex is the convex hull of the set of points forming the simplex, minus one. For example, the faces of a triangle are each of the three edges and the face of an edge is the set of two vertices that were connected to create that edge.

A *simplicial complex* X is a finite collection of simplices where any face of a simplex is itself a simplex in X . Further the intersection of any two simplices in X is either empty or a face of both simplices. Any simplex is uniquely determined by its vertices. For example, the object in Figure 1.3 is a simplicial complex consisting of the simplices

$$\{a, b, c, d, e, f, ba, ac, cb, bd, dc, ce, ef, ea, acb, cbd\}.$$

Notice the face of every simplex is included; for example the face of the 2-simplex acb is the set of 1-simplices $\{ac, cb, ba\}$. When simplices non-trivially intersect, their intersection is included

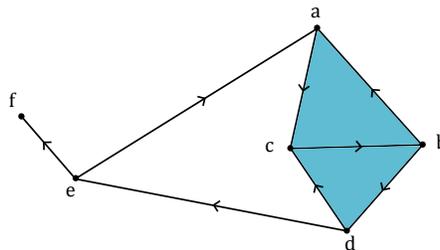


Figure 1.3: An example of a simplicial complex given by $\{a, b, c, d, e, f, ba, ac, cb, bd, dc, ce, ef, ea, acb, cbd\}$ with an imposed orientation.

in the complex and is itself a face of the two intersected simplices. Choices in the construction of a simplicial complex will be discussed later. We impose an ordering on the vertices which imposes an orientation on the simplex. In our example, the triangle acb may be given by any one of the equivalent names: $\{cba, bac, -cab, -bca, -abc\}$. The topology of a simplicial complex does not change under a consistent shuffling of the orientations of the simplices.

A p -chain is a subset of p -simplices in a simplicial complex. p -chains can also be thought of as formal sums, $c = \sum r_i \sigma_i$ where r_i is in $\mathbb{Z}/2\mathbb{Z}$, σ_i is a p -simplex in X . For example if our simplicial complex is a tetrahedron, each of the four triangle faces are 2-simplices. A 2-chain is any subset of these triangles. Similarly subsets of the edges and vertices form 1-chains and 0-chains respectively.

The set of p -chains of a simplicial complex form a p -chain group, called C_p . In fact, C_p is a free abelian group. It should be noted that when adding p -chains, the duplicate p -simplices cancel out. In our example the p -chain groups are

$$C_0(X) = \{\lambda_a a + \lambda_b b + \lambda_c c + \lambda_d d + \lambda_e e + \lambda_f f \mid \lambda_i \in \mathbb{Z}\} = \mathbb{Z}^6$$

$$C_1(X) = \{\lambda_{ac} ac + \lambda_{cb} cb + \lambda_{ba} ba + \lambda_{dc} dc + \lambda_{bd} bd + \lambda_{de} de + \lambda_{ef} ef + \lambda_{ea} ea \mid \lambda_{ij} \in \mathbb{Z}\} = \mathbb{Z}^8$$

$$C_2(X) = \{\lambda_{acb} acb + \lambda_{bdc} bdc \mid \lambda_{ijk} \in \mathbb{Z}\} = \mathbb{Z}^2$$

Now we define the boundary operator on chain groups. The *boundary* of a p -simplex is the set of $(p-1)$ -simplices' faces. Mathematically this is a formal sum of $(p-1)$ -simplices in the simplicial complex. Starting with 0-simplices, the boundary of a vertex is zero. The boundary of an edge is $\delta(ab) = b - a$. This extends to higher dimensions. For a k -simplex: $(v_0 v_1 \dots v_k)$ the boundary is the alternating sum $\delta(v_0 v_1 \dots v_k) = \sum_j = 0^k (-1)^j (v_0 v_1 \dots \hat{v}_j \dots v_k)$ where $(v_0 v_1 \dots \hat{v}_j \dots v_k)$ is the $k-1$ -simplex with the \hat{v}_j vertex removed. This can be extended to the entire chain group by linearly extending the operator on all chains. The boundary of a p -chain is the sum of the boundaries of its simplices modulo 2. That is, faces shared by an even number of p -simplices will cancel out.

Let's look at the boundary operator on a path in our example:

$$\delta(ac + cb + bd) = \delta(ac) + \delta(cb) + \delta(bd) = (c - a) + (b - c) + (d - b) = -a + d$$

and if we take the boundary of this result,

$$\delta(-a + d) = -\delta(a) + \delta(d) = 0$$

In general $\delta_n \circ \delta_{n+1}$ is zero, that is, the boundary of a boundary is zero. If the path is a loop, then the boundary of the path is zero. This is noteworthy if we are looking for “holes” in our simplicial complex.

Taking the boundary of a simplicial complex is a group homomorphism δ_p from C_p to C_{p-1} . In 3-dimensional space, this gives us a mapping like this:

$$\cdots \rightarrow \emptyset \rightarrow C_3 \xrightarrow{\delta_3} C_2 \xrightarrow{\delta_2} C_1 \xrightarrow{\delta_1} C_0 \xrightarrow{\delta_0} \emptyset$$

which is a chain complex. In our example we get

$$0 \rightarrow \mathbb{Z}^2 \xrightarrow{\delta_2} \mathbb{Z}^8 \xrightarrow{\delta_1} \mathbb{Z}^6 \xrightarrow{\delta_0} \mathbb{Z} \rightarrow 0$$

p -cycles are p -chains in the kernel of the boundary operator. The p -chains form a subspace Z_p of C_p . Z_p is by definition the kernel of δ_p . The p -chains that form the boundary of $(p + 1)$ -chains, are called p -boundary-cycles. In other words, the p -boundaries are the image of δ . These p -chains form B_p , a subspace of C_p , and $B_p \subset Z_p$. B_p is by definition the image of the $(p + 1)$ -boundary map. This can be visualized in Figure 1.4.

Geometrically, holes are loops which don't arise as boundaries of higher dimensional simplices. Algebraically, this is the quotient group. The p -th simplicial homology group of X is the quotient group

$$H_p = Z_p / B_p = \ker(\delta_p) / \text{Im}(\delta_{p+1}).$$

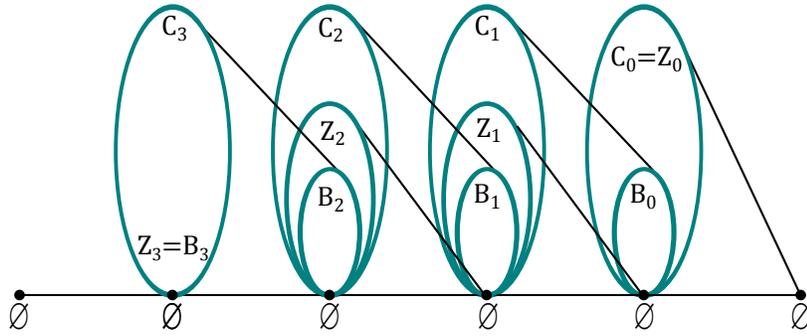


Figure 1.4: Schematically, the images of chain, cycle and boundary groups under the boundary operator. [32]

The p -th Betti number β_p is the rank of H_p and is intuitively the number of p -dimensional holes in the object.

1.3 Persistent Homology

Homology provides the algebraic structure needed to be able to begin to characterize geometric features of the data. However, to access homology, preprocessing of the data might need to occur. For example, if you are working with point cloud data sampled from some topological object, you may try to reconstruct an approximation by connecting proximate points with a simplicial complex. The homology of the resulting simplicial complex provides a snapshot of the homology of the original object. However, because this depends on a choice of threshold, homology is scale dependent in this case and provides limited information. Tracking how the homology changes as the proximity parameter increases gives a picture of the topological features that persist across many thresholds, this idea is called persistent homology [11, 30, 46].

Persistent homology is an algebraic method of computing topological features of spaces [30, 54]. It finds its utility in describing the multiscale topological structure of smooth functions, point cloud data or other geometric objects. What follows is a brief overview of the main concepts of persistence. ([31, 46, 54, 114] are good introductions to persistent homology.)

Consider a parameterized family of filtered spaces:

$$0 \subseteq \mathbb{X}_1 \subseteq \mathbb{X}_2 \subseteq \dots \subseteq \mathbb{X}_{\epsilon_m} \subseteq \mathbb{X}$$

equipped with the inclusion map [46]. The inclusion map on spaces $\mathbb{X}_i \subseteq \mathbb{X}_j$ for $i < j$ induces a homomorphism on the homology groups $\varphi_k^{i,j} : H_k(\mathbb{X}_i) \rightarrow H_k(\mathbb{X}_j)$ for each dimension k .

Rather than considering the homology of individual spaces, consider the homology of induced maps. Homology classes can then be described by the index at which they are born and the index at which they die. A homology class $\gamma \in H_k(\mathbb{X}_i)$ is born at \mathbb{X}_i if it is not in the image of $\varphi_k^{i-1,i}$ and dies entering \mathbb{X}_j if $\varphi_k^{i,j-1}(\gamma)$ is not in the image of $\varphi_k^{i-1,j-1}$ but $\varphi_k^{i,j}(\gamma)$ is in the image of $\varphi_k^{i-1,j}$. A homology class persists if its image under the induced map is nonzero. The difference between these two indices gives the persistence of the homology class. The multiset of birth-death pairs for each homological dimension can be represented in a barcode or a persistence diagram.

A common and concrete way to create this parameterized filtration is by building simplicial complexes on point cloud data. Given point cloud data of a sampled space, connect proximate points according to a chosen scale parameter by building a simplicial complex by putting simplices on top of points within the proximity parameter ϵ_i . The homology of the simplicial complex only changes at a discrete set of proximity parameters. We ensure that our choices of ϵ_i 's are between such thresholds. This creates a parameterized nested sequence with the natural inclusion map.

$$0 \hookrightarrow \mathbb{X}_{\epsilon_1} \hookrightarrow \mathbb{X}_{\epsilon_2} \hookrightarrow \dots \hookrightarrow \mathbb{X}_{\epsilon_m} \hookrightarrow \mathbb{X}$$

Applying the homology functor to the filtration, we get a persistence module.

$$H_*(\mathbb{X}_{\epsilon_1}) \rightarrow H_*(\mathbb{X}_{\epsilon_2}) \rightarrow \dots \rightarrow H_*(\mathbb{X}_{\epsilon_m}) \rightarrow H_*(\mathbb{X})$$

The information in the persistence module is summarized in a persistence diagram or barcode (which is shown in Figure 1.5). The persistence diagram is the multiset of birth-death pairs

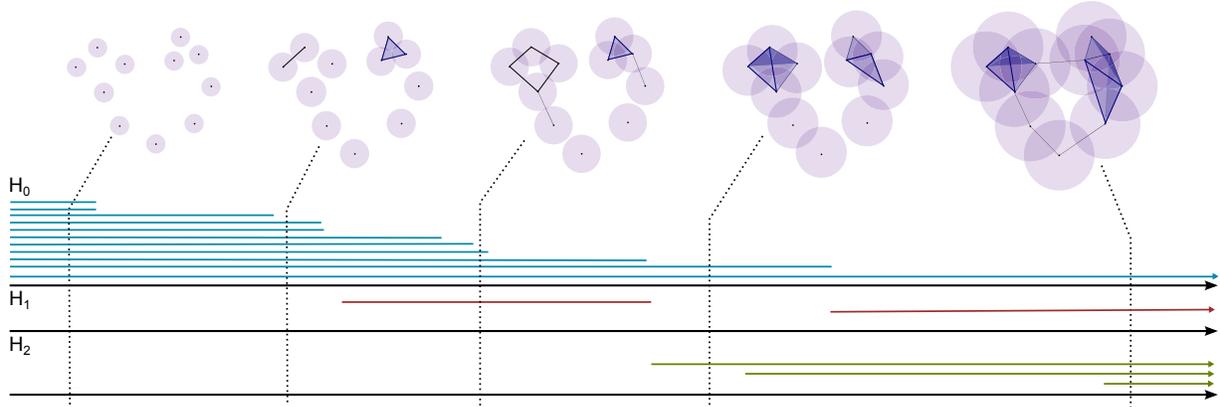


Figure 1.5: A Vietoris-Rips complex built on point cloud data (top) in \mathbb{R}^3 and the corresponding barcodes for H_0 , H_1 and H_2 for the entire filtration (bottom).

describing homological features. It should be noted that if we impose a partial ordering on the vertices, these computations can be done with matrix algebra. See [30] for details.

1.4 Building Simplicial Complexes

To compute persistence homology of a set of point cloud data, one might be to build a filtration of simplicial complexes on the data. This becomes a combinatorial problem, because a simplicial complex formed from a set of discrete data points is far from unique. There are several common methods for building a simplicial complex from discrete data which we will introduce before we discuss choosing an appropriate simplicial complex to represent the data contained in the point cloud.

Let X be a finite set of points in \mathbb{R}^d . $B_x(r)$ is the ball with radius r and center x . To build the associated Čech complex, add a d -simplex when there is a common point of intersection of all d $(\epsilon/2)$ -balls.

$$\check{C}ech(r) = \{\sigma \subseteq X \mid \bigcap_{x \in \sigma} B_x(r) \neq \emptyset\}$$

The resulting simplicial complex depends on the choice of scale parameter. For $s < t$ we have $\check{C}ech_s(X) \subseteq \check{C}ech_t(X)$ so a series of Čech complexes with increasing radii induces a filtration X with the natural inclusion map. The Čech complex does not necessarily have a geometric realization in \mathbb{R}^d , but we can treat it as an abstract simplicial complex without any problem.

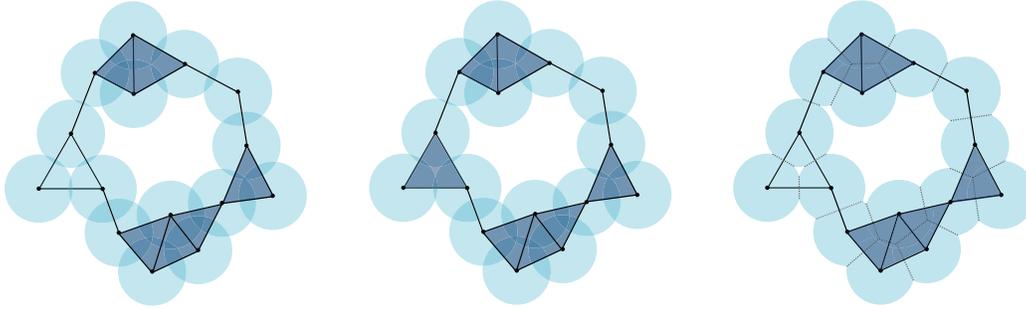


Figure 1.6: The Čech, Vietoris-Rips and alpha complexes respectively of a noisy sampling of an annulus. The difference between the two complexes is highlighted by the three points equidistant apart from each other. In the Čech complex, they are connected by 1-simplices, but not a 2-simplex. In the VR complex, we include the 2-simplex. We can see $\check{C}ech(\epsilon) \subseteq VR(\epsilon) \subseteq \alpha$.

Storage of the Čech complex is expensive and in the case of larger data sets, can be computationally prohibitive. The Vietoris-Rips complex is similar to the Čech complex, but less memory intensive and therefore more readily computable. To build the Vietoris-Rips (VR) complex, add a d -simplex when all of the vertices have nonempty pairwise intersections.

$$VR(r) = \{\sigma \subseteq X \mid \text{diam} \sigma \leq 2r\}$$

The resulting simplicial complex depends on the choice of scale parameter. For $s < t$ we have $VR_s(X) \subseteq VR_t(X)$ so a series of VR complexes with increasing radii also induces a filtration X with the natural inclusion map.

The distinction between these a Čech complex and Rips complex on the same point cloud is seen clearly in the example of a set of three vertices that are equidistant apart. There is a small range of radii for which, the balls centered at each vertex intersect pairwise, but do not mutually intersect. In the Čech simplicial complex there will be edges connecting the vertices, but no 2-simplex. The Vietoris-Rips complex will include the 2-simplex, seen on the left side of the complex in Figure 1.6.

It is not immediately clear how accurately the Vietoris-Rips complex actually captures the topological features of a discrete sampling of a topological object. A Vietoris-Rips complex built on a point cloud sampled from a manifold can include features that are artifacts of the complex

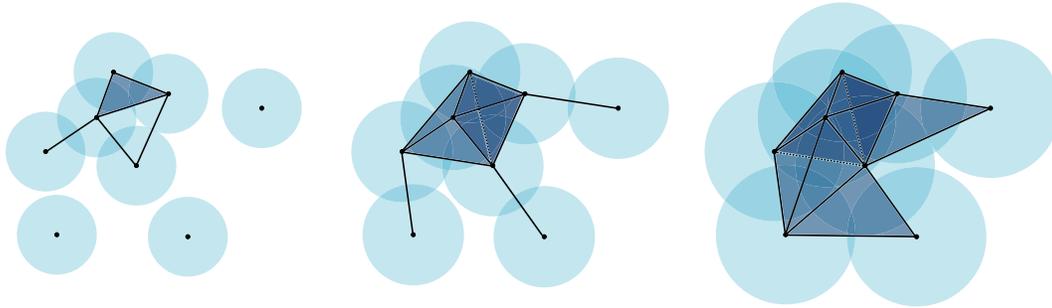


Figure 1.7: Let X be a point cloud in \mathbb{R}^3 . Three Čech complexes formed at increasing values of ϵ are shown. Each simplicial complex includes into the one to the right, which is formed by increasing the radius of the balls. We can see that the topological signature of our complex changes with each different value of ϵ .

and are not representative of the topology of the manifold from which points were sampled [47]. However, the Čech complex is a reasonable representation of the homology of the underlying object due to the Nerve Lemma in algebraic topology, which claims that given a covering of the space with balls and the associated Čech complex associated to the covering is homotopy equivalent [31].

Further, can “squeeze” a VR complex between two Čech complexes with appropriately chosen radii. [31]

$$\check{C}ech(r) \subseteq VR(r) \subseteq \check{C}ech(\sqrt{2}r)$$

This allows us to proceed with building a filtered sequence of spaces using either the Čech complex or the VR complex. We will see in Chapter 5 that in different applications, it is advantageous to use one complex over the other.

1.5 Discrete Morse Theory

If we have a point cloud, building an appropriate filtration of simplicial complexes is a natural approach. However, if we wish to compute homology a surface $u(x, y, \cdot)$ which is the solution to a PDE, we need to consider filtrations on functions. This will take the form of filtrations on cell complexes first, and then we will extend to a more computable version.

Classical Morse theory (see [77]) allows one to study the topology of a manifold by studying differentiable functions on the manifold. Given M a compact differential manifold and a function $f : M \rightarrow \mathbb{R}$ which is smooth with critical points that have a non-singular Hessian (this is called a Morse function) consider the negative gradient flow of f . We use the function f to build a cell complex. A cell complex consists of cells (e.g. vertices, edges, and quads) and boundary maps that represent the neighborhood relationship of these cells. Cells in the Morse complex will be defined by regions that exhibit the same low behavior. Following [100], think of a landscape with hills and valleys. The regions where rain flows into the same basin correspond to cells in the Morse complex. They are divided by ridge lines which pass through saddle points. The isolated points where multiple such basins touch are maximas. Mathematically these are the stable manifolds of the critical points. Think of the unstable manifolds in a similar way, partitioning the surface by common points of origin of flow rather than termination. The intersection of the stable and unstable manifolds is the Morse-Smale complex [100]. This encodes a gradient or vector field in the combinatorial structure of a cell complex [96].

We can define a filtration on a well-behaved function on a compact manifold from the lower level cuts (that is, the sublevel sets) of f . From Morse theory, we know that the topology of the sublevel sets only change when we pass through a critical point. It has been shown that the persistent homology of the sublevel set filtration is the same as the persistent homology of the abstract cell complex defined by critical points and gradient flow lines [98].

The function f is called a filtering function. Cohen-Steiner et al. [12] proved that for any two filtering functions f and g , the difference of their persistence is bounded by the L^∞ norm of their difference. From this stability theorem, we can say that when the persistence of two functions is sufficiently different, the functions themselves are different. That is, we can use persistence as a signature of functions!

Robin Forman extends Morse theory to a combinatorial analog, called discrete Morse theory in [40, 41], considering functions on CW complexes. (Cells in a CW complex are homeomorphic to Euclidean balls.) A discrete Morse function assigns a single value to each cell, which imposes

partial ordering on cells. Robins et. al describe how to construct a discrete Morse function on a discretization of a manifold so that the critical cells exactly match changes in topology of a sublevel set filtration of cubical complexes [100]. We can think of this as combinatorial gradient paths between critical cells. We first define cubical complexes.

1.5.1 Cubical Homology

Cubical complexes allow us to avoid triangulating the space and computing persistence on the resulting simplicial complex (as has been done, for example in [31]). Wagner, Chen and Vućini propose an efficient algorithm for the computation of persistent homology of cubical data that can be extended to higher dimensions. [117]

Cubical complexes are highly computable [79]. They are similar to a simplicial complex, but instead of nodes, edges, triangles and tetrahedrons, cubical complexes consist of nodes, edges, pixels squares and cubes. For example, see Figure 1.8. Consider the solution surface $u(x, y, t_i)$ for some fixed value of time t_i on an $n \times m$ spatial domain (i.e., the solution is given as an $n \times m$ matrix). The value of each pixel is interpreted to be the value of a vertex of the complex. Building cubical complexes on the sublevel sets of the height function produces a filtration. The filtration is built as follows: for a threshold ϵ a vertex is included if the function value at that node is less than or equal to ϵ . An edge between nodes is included if both vertices are included, and a square is included if all four associated vertices are included. In the 2D framework, each vertex can connect to four neighbors, this is called 4-connectivity. In general, in dimension d , we use $2d$ -connectivity. When all possible components of the cubical complex are included, we will have nm nodes, $(n-1)m + n(m-1)$ edges and $(n-1)(m-1)$ squares.

In the same way as was done for simplicial complexes, we can define the boundary map, p -cycles, chain complexes and homology groups. The homology of the resulting cubical complex will change for a finite set of ϵ values, which can be used to parametrize the filtration. We compute persistent homology on the resulting filtration. Conceptually, this gives a notion of the magnitude of peaks and valleys present in a surface.

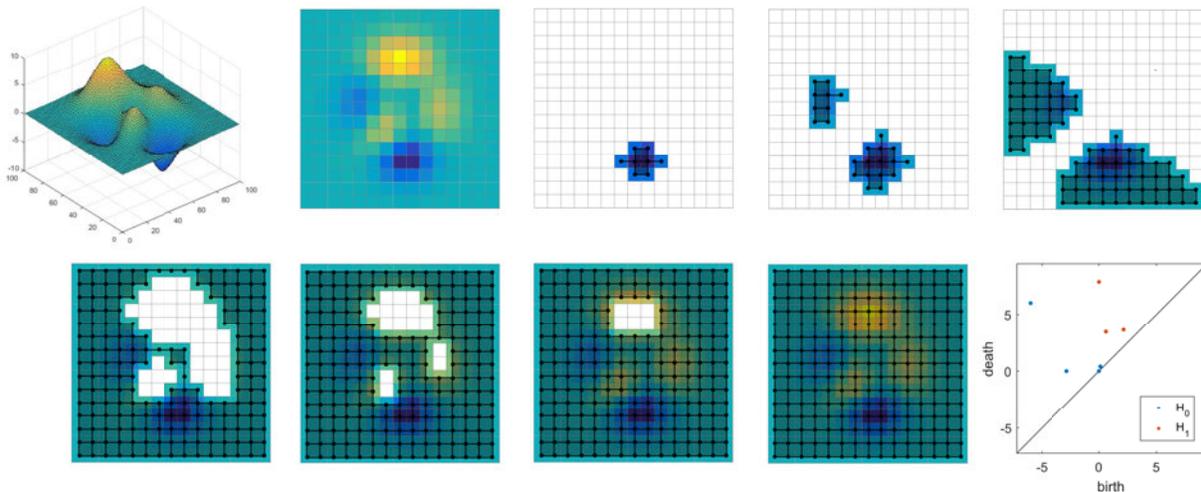


Figure 1.8: Example of a filtration of cubical complexes built on the matrix containing surface height information on the left (blues are low values and yellows are high values).

As an analogy, think of the sea and the surface of landmass. At some sea-level, there are components that are under the water, there are islands of land, and on these islands, there may be valleys whose lowest point falls below the sea level and so these valleys fill with water. (Here we are assuming the surface of all water is at a fixed height, a small diversion from real topographical land features.) If we restrict our vision to the land that is under the surface of the water (sublevel sets), the islands cut holes in the land that we can see, and the lakes introduce additional components of underwater land. We could also think of the superlevel set, that is, all of the land that is above the surface of the water. Here the islands are each their own component, and the lakes cut holes in these components.

We would like to study topological structure present in the solutions to PDEs. The solution, to a two-dimensional spatio-temporal system will be a surface $u(x, y, \cdot)$. In most cases, the solution is not analytic, so numerical methods are used in simulation. The result is a discretized solution on a spatial grid (of desired resolution). Cubical complexes are a natural choice for a filtration here and we can think of it as a discretized version of the sublevel set filtration of a function [11]. There is a choice in using discrete data on a grid whether to treat the function values as vertices or cubes in the cubical complex. We will treat them as cubes, however the two approaches are dual and so they encode the same topological information [100]. In higher

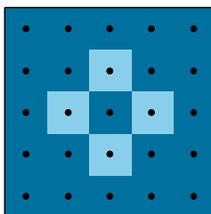


Figure 1.9: A motivating example for choosing 4-8 connectivity. The value of each discrete point is indicated by the color of the square on which it sits. If we use 4-neighbor connectivity for all points, then the light points are totally disconnected, but still partition the dark points into two components. If we use 8-neighbor connectivity for all points, then the light points form the discrete analog of the Jordan curve theorem but do not separate the inside dark point from the rest. [62]

dimensions, this method has been used to study plant root structure [34], and x-ray micro-CT data of rock cores [26].

Cubical complexes have been used to study grayscale images, for example see [26, 34, 100], by creating a filtration of thresholded images. We think of a numerical simulation of the solution of a PDE in the same way. Kovalsky [63] shows that a cubical complex is the only topologically consistent representation of a thresholded image. However, the resulting persistence diagrams do not satisfy the fundamental symmetries that are implied by Lefschetz and Alexander duality [34]. We carefully adjust our notion of connectivity so that we achieve a representation that maintains fundamental topological symmetries.

For a binary representation of a thresholded image, one assigns a value of 1 if a pixel is below some threshold and 0 if it is above the threshold. In order to construct cubical complexes of a thresholded image, the values of the 4 neighboring pixels were considered for. However, this leads to challenges with connectivity and duality if we build two cubical complexes, one on the vertices with value 1 and one on the vertices with value 0. Instead, we create a dual complex with 8 neighbor connectivity on the vertices with value 0 [34]. This was a basic convention in the field of digital topology, which was concerned with 2D and 3D digital images and their topological properties. A nice survey is given by Kong and Rosenfeld [62].

In order to be discrete analog of the Jordan curve theorem [62] (which states that every simple closed curve divides the plane into an interior connected component and an exterior con-

nected component) we use 4 neighbor adjacency to construct complexes of sublevel sets and 8 neighbor adjacency to construct complexes of superlevel sets. This allows us to avoid paradoxes such as the one pictured in Figure 1.9. We center a vertex on each pixel, the value of which is indicated by the color of the pixel on which it sits. If we use 4-neighbor connectivity for all points, then the light points are totally disconnected, but still partition the dark points into two components. If we use 8-neighbor connectivity for all points, then the light points create a discrete analog of a Jordan curve, but the dark points are connected as a single component as well. The Jordan Curve Theorem generalizes to Lefschetz duality [34]. Lefschetz duality connects the homology of a manifold with boundary to the relative homology of the manifold and its boundary. In the next section, we briefly introduce relative homology.

1.5.2 Relative Homology

Produce a filtrations of complexes on a data set \mathbb{X} , given by $0 \subseteq \mathbb{X}_1 \subseteq \mathbb{X}_2 \subseteq \dots \subseteq \mathbb{X}_{\epsilon_m} \subseteq \mathbb{X}$. Applying the homology functor, we get the persistence module:

$$0 = H_k(\mathbb{X}_0) \rightarrow H_k(\mathbb{X}_1) \rightarrow \dots \rightarrow H_k(\mathbb{X}_0)$$

However, the module is not necessarily exact. Each death will correspond to a unique birth, but not every birth will correspond to the death of a feature. $H_k(\mathbb{X}_0)$ possibly ends with nontrivial homology groups that never die [33]. The consequence here is that we lose valuable information on the duration of such classes. We follow [13] to extend the sequence of homology.

$$0 = H_k(\mathbb{X}_0) \rightarrow H_k(\mathbb{X}_1) \rightarrow \dots \rightarrow H_k(\mathbb{X}_0)$$

Duality implies that in this formulation, we get each birth-death pair twice.

Relative homology makes precise the idea of the homology of X modulo a subspace A [33]. Given a space X and a subspace $A \subseteq X$, define $C_n(X, A)$ to be the quotient group $C_n(X)/C_n(A)$. The boundary map on $\delta : C_n(X) \rightarrow C_n(X)$ will also map $C_n(A) \rightarrow C_n(A)$ and induces the quo-

tion boundary map $\delta : C_n(X, A) \rightarrow C_{n-1}(X, A)$, which gives a sequence of boundary maps. The homology groups of this chain complex are $H_n(X, A)$. Elements in $H_n(X, A)$ are represented by n -chains, $\alpha \in C_n(X)$ with the boundary $\delta\alpha \in C_{n-1}(A) \subset C_{n-1}(X)$. Think of this as an n -thing whose boundary, which is an $n-1$ thing, lives in A [54]. That is, relative homology is concerned with the part of X outside A .

Relative homology groups fit into the long exact sequence

$$\dots \rightarrow H_n(A) \rightarrow H_n(X) \rightarrow H_n(X, A) \rightarrow H_{n-1}(A) \rightarrow H_{n-1}(X) \rightarrow \dots \rightarrow H_0(X, A) \rightarrow 0$$

and measures the difference between groups $H_n(X)$ and $H_n(A)$ [31]. Here, every class that is born also dies. Duality implies that in this formulation, we get each birth-death pair twice for homology groups that appear in the original persistence module of our filtration [12]. In fact this duality is exploited in [52]. All persistent homology computations in this thesis are done using [110] and [52]. We return now, with more robust topological properties, to computing persistence of a function.

1.5.3 Persistent Homology of a Function

In the functional setting, we will compute persistence using a discrete sublevel set Morse filtration on the height. That is, we build a sublevel set filtration [30]. Let $f : \mathbb{X} \rightarrow \mathbb{R}$, where \mathbb{X} is a geometric object and f is a smooth function. Consider the nondegenerate critical points, that is, when $f'(x) = 0$ and $f''(x) \neq 0$. For each $\epsilon \in \mathbb{R}$ the sublevel set is $\mathbb{R}_\epsilon = f^{-1}(-\infty, \epsilon]$. Increasing ϵ produces a filtration of sublevel sets:

$$f^{-1}((-\infty, \epsilon_1]) \subseteq f^{-1}((-\infty, \epsilon_2]) \subseteq \dots \subseteq f^{-1}((-\infty, \epsilon_m]).$$

Here we are concerned with the connectivity of sublevel sets, which changes only as ϵ passes through a critical value. Computing the homology of this filtration involves keeping track of

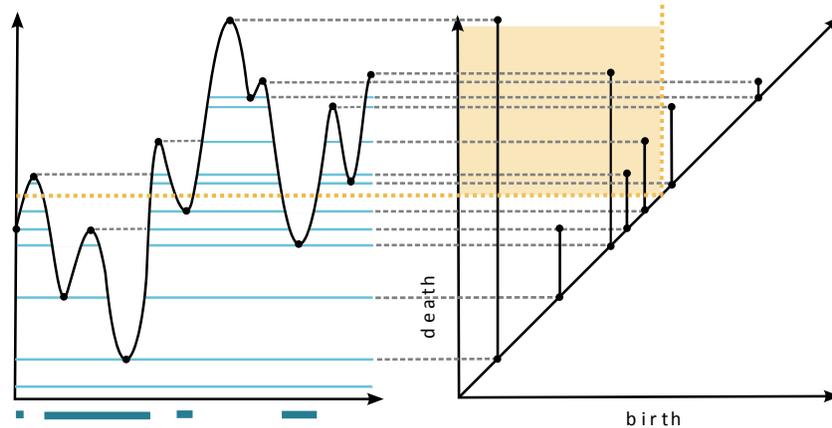


Figure 1.10: Persistence of the sublevel sets of a one-dimensional function. The gold line indicates a choice of a threshold. Below the function in teal are the sublevel sets associated with the height of the gold line. Tracing this height to the persistence diagram, the four components can be counted as points in the shaded gold region.

the filtration parameter ϵ for which connected components appear and merge, that is, when a feature is born and dies. See Figure 1.11.

There is a connection here to the point cloud perspective of persistence. Given $X \subseteq \mathbb{R}^n$, we define the distance function $d_X : \mathbb{R}^n \rightarrow \mathbb{R}$ to be

$$d_X(a) = \inf_{x \in X} \|a - x\|$$

This function encodes the Čech complex persistent homology as functional persistent homology [114]. This function is illustrated for a 1-dimensional point cloud in Fig. 1.11, and extends naturally to higher dimensions.

The higher dimensional analog of this method is computing a discrete sublevel set Morse filtration on the height function of a discretized samplings of a surface. We have a function value associated to each point on a grid. From the function values, we build a sublevel set filtration and compute persistent homology. See Figure 1.8 for example. We find numerous applications of this computational technique. Ultimately this allows us to exploit topological features of data

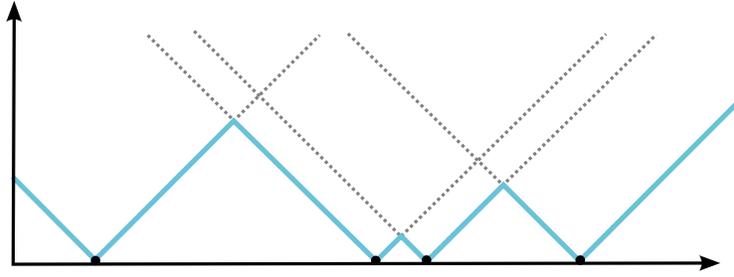


Figure 1.11: Consider a 1-dimensional point cloud. We draw the distance function as cones emanating out from each point. The infimum of these distances is the function of distances to the entire set. This encodes Čech homology in the functional setting.

to study and characterize complex patterns. These patterns often arise as solutions to PDEs, and are dependent on a number of parameters, so we need a good way to compare the resulting persistence diagrams. We discuss several techniques to quantitatively compare persistence diagrams in the next section.

1.6 Persistence Images as a Classification Tool

Information about topological features is captured in persistence diagrams. To enable quantitative comparisons, the space of persistence diagrams can be endowed with a metric structure. The most common metrics are the p -Wasserstein distance and the Bottleneck distance, which are both stable with respect to small perturbations of the underlying data [12].

The p -Wasserstein distance, defined between two persistence diagrams B and B' , is given by

$$W_p(B, B') = \left(\inf_{\gamma: B \rightarrow B'} \sum_{u \in B} \|u - \gamma(u)\|_\infty^p \right)^{1/p},$$

where $1 \leq p < \infty$ and γ ranges over bijections between B and B' . Taking the limit as $p \rightarrow \infty$ gives the *bottleneck distance*.

$$W_\infty(B, B') = \inf_{\gamma: B \rightarrow B'} \sup_{u \in B} \|u - \gamma(u)\|_\infty,$$

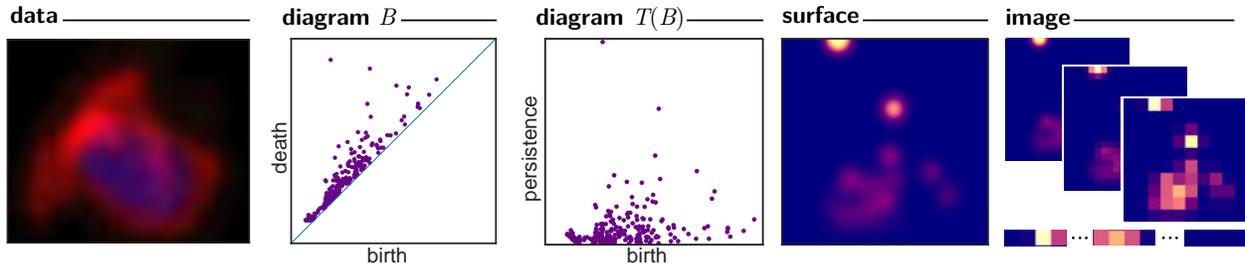


Figure 1.12: Illustrated above is the pipeline for generating a persistence image. The persistence diagram is computed from data, transformed to birth-persistence coordinates, weighted 2-dimension Gaussian distributions are placed on top of each persistence point to create a surface, and the surface is gridded and integrated to create a stable matrix representation of the persistence diagram. (Figure from Adams et al. [2])

which can be thought of as minimizing the “cost” of pairing points in two different persistence diagrams (including the option of pairing as many times as needed with the diagonal). While there are some nice stability properties [14], these distances can be computationally expensive as the number of points increases. While recent strides in algorithms [61] have significantly decreased computation time for the bottleneck distance, there are machine learning algorithms that don’t just require a metric space structure, but a vector space structure.

There have been an number of attempts recently to create representations of persistence diagrams that allow for wider use of machine learning techniques. Several representations include persistence landscapes [9], which provides a functional representation of a persistence diagram with a unique mean, using tropical coordinates [115], which creates a feature-vector representation that can be used for machine learning tasks, and kernel methods [97], which involves taking the sum of positive Gaussians centered on each persistence point with a negative Gaussian centered at the reflection of each point over the diagonal. While these techniques are useful in certain contexts, here we will use our representation, *persistence images*, [2] because it provides a stable vector representation that is interpretable and flexible.

We create a persistence image of a persistence diagram B , in the usual birth-death coordinate system. Consider $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ the change of coordinates to a birth-persistence coordinate system, that is $T(x, y) = (x, y - x)$. Then $T(B)$ is the multiset of birth-persistence points. Let $\phi_u : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a differentiable probability distribution with mean $u = (u_x, u_y) \in \mathbb{R}$. A common

choice for ϕ is the normalized symmetric Gaussian $\phi_u = g_u$ with mean u and variance σ^2 defined as

$$g_u(x, y) = \frac{1}{2\pi\sigma^2} e^{-[(x-u_x)^2 + (y-u_y)^2]/2\sigma^2}.$$

Let $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a nonnegative weighting function that is zero along the horizontal axis, continuous, and piecewise differentiable. Such a weighting function ensures stability. (See [2] for full proof.) Depending on prior knowledge of data, one may wish to use a weighting function such as a sigmoidal function that gives little weight to points close to the diagonal, which in some cases is thought of as noise. A simple choice for a weighting function which only depends on the persistence coordinate y is a piecewise linear function, defined as follows:

$$w_b(t) = \begin{cases} 0 & \text{if } t \leq 0, \\ \frac{t}{b} & \text{if } 0 < t < b, \text{ and} \\ 1 & \text{if } t \geq b. \end{cases}$$

In our applications, we use $f(x, y) = w_b(y)$, where b is the persistence value of the most persistent feature in all persistence diagrams under consideration, chosen for each homological dimension.

The process of converting a persistence diagram to a persistence image is seen in Fig. 1.12. The first step in the transformation of a persistence diagram B is to create a persistence surface, $\rho_B(z)$ by summing the weighted Gaussians at each point in the transformed diagram $T(B)$. That is:

$$\rho_B(z) = \sum_{u \in T(B)} f(u)\phi_u(z).$$

Next, discretize a relevant subdomain of the surface by overlaying a grid of the desired resolution and integrating over each box in the discretization: $I(\rho_B)_p = \iint_p \rho_B dydx$. The value of the integral gives value of each pixel in the image representation. In a number of experiments, it has been seen that various classification results are fairly robust to a large range of resolutions.

Often a resolution of 10×10 is sufficient to capture variations between persistence diagrams in classification tasks.

Persistence images is a stable vector representation of each persistence diagram. We will use this tool to leverage on the topological structure of solutions to PDEs for classification tasks.

Chapter 2

Persistent Homology and Unimodal Maps

Given discrete sampling of data in a time series, it can be difficult to detect whether data comes from a deterministic or stochastic process, especially when there is not a visually discernible pattern. In other words, is the data chaotic or random? This simple question turns out to be more challenging than it initially appears and has intrigued mathematicians and scientists for quite some time. Efforts to better understand chaotic systems has led to major contributions to the field of dynamics. A classic signature of chaos is a sensitive dependence on initial conditions. Two orbits with arbitrarily close initial conditions will diverge exponentially. If we start with a discrete sampling of data, rather than the map describing the mechanism, this signature is harder to detect. In this chapter, we introduce a method to detect an underlying deterministic system (of a specific type) given a discrete sampling of chaotic data.

With the assistance of computers, scientists have been able to observe chaotic behavior in contexts that were unexpected. In 1963, Edward Lorenz noticed a sensitive dependence on initial conditions while running simulations on an atmospheric model [72]. His idea came to be known in popular culture as the “butterfly effect”. His finding challenged classical understanding of natural systems and in the next several decades, his work became foundational in chaos theory. In 1976, biologist Robert May championed the notion that simple one-dimensional nonlinear systems can display a wide spectrum of complex dynamic behavior, including periodic orbits and chaotic orbits [74]. Metropolis, Stein, and Stein [76] described characteristics periodic orbits in a large class of one-dimensional systems. A few years later, Li and Yorke proved that the presence of a period-3 orbit implied the existence of chaos [71].

Strikingly, chaotic systems displayed a type of universal behavior if they were viewed with the right tools [99]. Feigenbaum discovered that a large class of maps displayed a geometric universality in a fractal pattern present in the bifurcation diagrams. He also determined universal scaling invariance that scales the parameter values of the onset of period doubling cas-

cales [38] which are eigenvalues of a renormalization transform [99]. This discovery led to a surge in interest in nonlinear dynamics.

Nearly a century earlier, Poincaré had asserted that the most important governing features of a system are the things that you can't see—the unstable periodic orbits. By studying these orbits, one can learn a great deal about the behavior of a nonlinear system. Mathematicians such as Gilmore began studying topological invariants of dynamical systems [48].

There is, in fact, topological structure in dynamical systems, and there has been growing interest in recent years in studying chaotic dynamical systems from a topological point of view. Topological invariants can be extracted from chaotic data that describe the stretching and squeezing mechanisms which create strange attractors [48]. These mechanisms organize the unstable periodic orbits in specific way; the stretching mechanism causes sensitivity to initial conditions and the squeezing mechanism is responsible for recurrent non-periodic behavior. This leads to a strange attractor (displaying chaotic behavior) that has a self-similar structure and is represented as a branched manifold (also called a template).

Qualitative behaviors studied from the point of view of topology has led to several fixed point theories [78] such as Conley index theory that seek to study dynamical system from a topological perspective. These techniques rely heavily on knowing the analytic form of the dynamical system, or at a minimum having access to periodic orbits. Work has been done to decide if data is chaotic by, for example, estimating Lyapunov exponents² by using a delay embedding to reconstruct the attractor. See [84], for example, for a geometric approach to computing Lyapunov exponents.

This chapter aims to do the following: to describe and motivate an interesting pattern that arises in when persistence is computed on several examples of one-dimensional dynamical systems that display periodic and chaotic behavior, and to provide a connection with universal behavior of a larger class of dynamical systems.

²Lyapunov exponents measure how quickly two points that are initially close diverge under the given map. A positive Lyapunov exponent indicates chaotic behavior.

2.1 Discrete Dynamical Systems

We focus our attention on one-dimensional discrete dynamical systems. Let $f : \mathbb{R} \mapsto \mathbb{R}$ be a real-valued function, called a map. Starting from an initial point, x_0 , f maps to the sequence $\{x_0, f(x_0), f^2(x_0), \dots, f^n(x_0) \dots\}$, where $f^n(x_0)$ represents the n^{th} composition of f with itself. The set $\{f^n(x_0) | n \in \mathbb{N}\}$ is called the *orbit* of x_0 under f . We can also write this map as a first order difference equation

$$x_{n+1} = f(x_n),$$

where $x_n = f^n(x_0)$, the n^{th} iteration of the initial condition x_0 . First-order difference equations are simple dynamical systems that can display rich structure and interesting dynamics.

When $f(x^*) = x^*$, x^* is called a *fixed point*. Fixed points play an important role in understanding the dynamics. Similarly, when $f^n(x_0) = x_0$, x_0 is a *periodic point*. The period of such a point is n as long as $f^m(x_0) \neq x_0$ for any $0 < m < n$. A periodic point has a finite periodic orbit (of size n), where $f^n(x_m) = x_m$ for all x_m in the orbit. A point x_0 is *eventually periodic* if $f^n(x_m) = x_m$ for some n and x_m in the orbit of x_0 . In the same way a point x_0 is eventually fixed if $f^k(x_m) = x_m$, for some k and x_m in the orbit of x_0 . A set $U \subseteq \mathbb{R}$ is invariant if $f(U) \subseteq \mathbb{R}$.

We can classify fixed points in the following way, based on dynamics of nearby points. We call x^* an attracting fixed point for f if there exists some open neighborhood U of x^* such that if $x \in U$ then $f^n(x) \in U$ for all n and $f^n(x) \rightarrow x^*$ as $n \rightarrow \infty$. The neighborhood $(0, x_2^*)$ is such a neighborhood for the attracting fixed point x_1^* in Figure . A necessary and sufficient condition for the local stability of the equilibrium point is given by evaluating the first derivative of the difference equation evaluated at the fixed point indicated the stability. If x^* is a fixed point and $|f'(x^*)| < 1$, then x^* is an attracting fixed point [108]; that is, it is locally stable. Likewise, a fixed point x^* can be identified as a repelling fixed point if $|f'(x^*)| > 1$, which means the fixed point is locally unstable. If $|f'(x^*)| = 1$, then x^* is called indifferent or neutral and is neither attracting nor repelling [56].

It is easy to visualize these systems through graphical iteration using a tool called cobwebbing. Plot the difference equation $x_{n+1} = f(x_n)$ and the line $x_{n+1} = x$. Start at the point $(x_a, 0)$

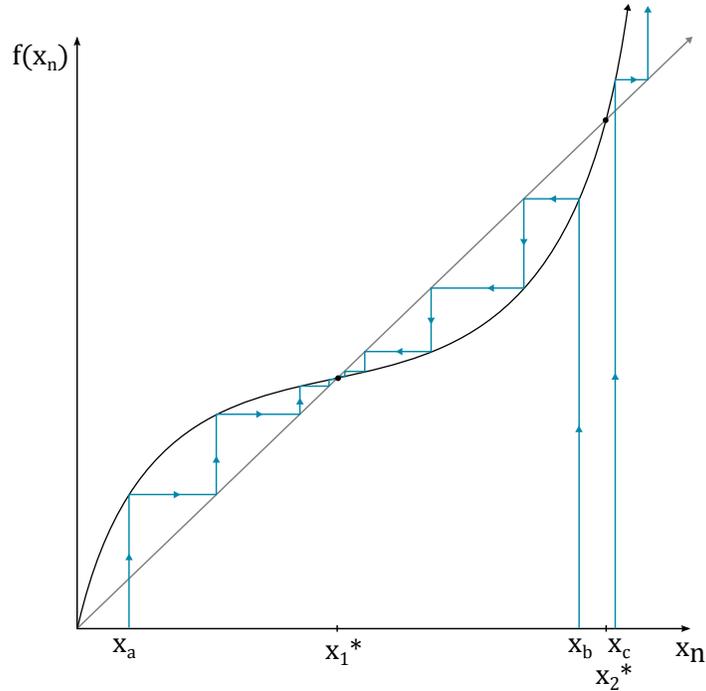


Figure 2.1: Several cobwebs trace the orbits of a few initial conditions. Notice 0 and x_2^* are repelling fixed points and x_1^* is an attracting fixed point.

and move vertically to the graph of f . Now we are at $(x_a, f(x_a))$ which is (x_a, x_{a+1}) . Now move horizontally back toward the diagonal, to (x_{a+1}, x_{a+1}) . Repeat this process, moving vertically to $(x_{a+1}, f(x_{a+1})) = (x_{a+1}, x_{a+2})$ and horizontally back to the diagonal at (x_{a+2}, x_{a+2}) . This process visually shows the orbit of a point and gives us an intuitive understanding of long term behavior. This is seen in Figure 2.1.

The ω -limit set of x under f is

$$\omega(x, f) = \{y \in \mathbb{R} \mid \text{there exists } n_1 < n_2 < \dots \text{ with } f^{n_i}(x) \rightarrow y\}$$

$\omega(x, f)$ may be finite, in which case, it contains a periodic limit cycle, or it may be infinite. Studying the ω -limit sets of a map will give insight into the dynamics that are present. Often the behavior of discrete dynamical systems depend on a control parameter appearing in the equation, that is, $f(x, r)$. We will see that varying the control parameter r can have a large effect on the dynamics of the system [56].

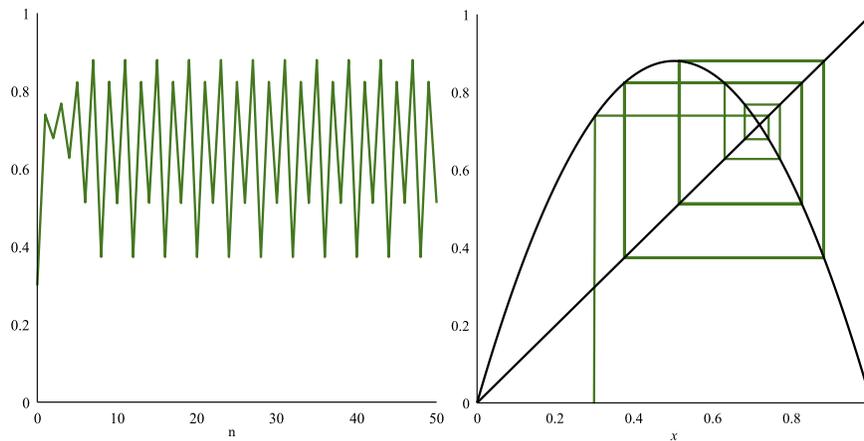


Figure 2.2: Cobwebbing shows the stability of the period 4-cycle for $r = 3.52$. The left shows the orbit of a point in time and the right shows the plot of the map and the cobwebbing corresponding to the orbit.

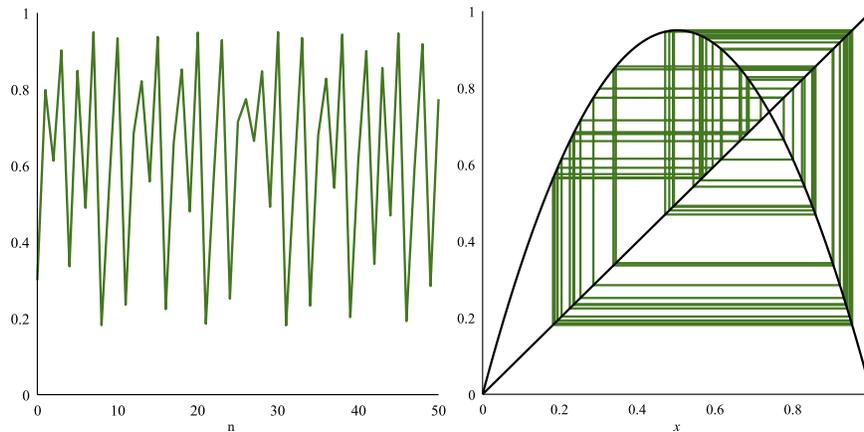


Figure 2.3: Cobwebbing displays chaotic behavior when $r = 3.8$. The left shows the orbit of a point in time and the right shows the plot of the map and the cobwebbing corresponding to the orbit.

2.1.1 The Logistic Map

Our first example will be the logistic map. Championed by biologist Robert May in 1976 [74] as clear example that simple nonlinear maps could have very complicated dynamics, the logistic map is one of the simplest and most well understood examples of a nonlinear discrete dynamical system. For various parameter values, it displays the most important features of low-dimensional chaotic behavior. The logistic map is given by

$$x_{n+1} = r x_n (1 - x_n).$$

This map is interesting because it provides a rich example to explore periodic regions, complex chaotic behavior and self similarity. The logistic map maps the unit interval $I = [0, 1]$ to itself for values of $r \in [0, 4]$ which means that I is forward invariant.

Fixed points of this map give some insight into the dynamics of the map. The fixed points for this map are $x_0^* = 0$ and $x^* = 1 - \frac{1}{r}$. This means that for all r , the origin is a fixed point and for $r \geq 1$, $x^* = 1 - \frac{1}{r}$ is in the unit interval. Stability is determined by $|f'(x^*)| = |r - 2rx^*|$, which means that the origin is stable for all $r < 1$ and unstable for $r > 1$. For the other fixed point to be stable, $|r - 2r(1 - \frac{1}{r})| = |2 - r| < 1$ must hold, implying that for $1 < r < 3$ this fixed point is stable and for $r > 3$ this fixed point is unstable.

At $r = 1$ we say that the fixed point at the origin splits or bifurcates in a transcritical bifurcation. That is, one fixed point will lose stability and the other will gain stability. The second fixed point (shown on the graph as x^*) remains stable as r increases until r reaches 3, at which point $f'(x^*) = -1$ and the map undergoes another bifurcation called a saddle-node bifurcation. A saddle-node bifurcation occurs when a single, neutral fixed point instantaneously splits into two fixed points, one attracting, or stable (called the node) and one repelling, or unstable (called the saddle).

Note that the unstable fixed point occurs on the border between the basin of attraction of the stable fixed point and the basin of attraction for infinity. We notice from the bifurcation diagram Figure 2.4) that for $3 < r < 3.449489728\dots$, the logistic map has a stable period 2-cycle.

Table 2.1: The parameter values for onset of the first several period 2^n -cycles for the logistic map.

Parameter	Cycle	Value
r_1	2	3
r_2	4	3.449489728...
r_3	8	3.544090359...
r_4	16	3.5644072661...
r_5	32	3.5687594195...
r_6	64	3.5696916098...
\vdots		
r_∞		3.5699456...

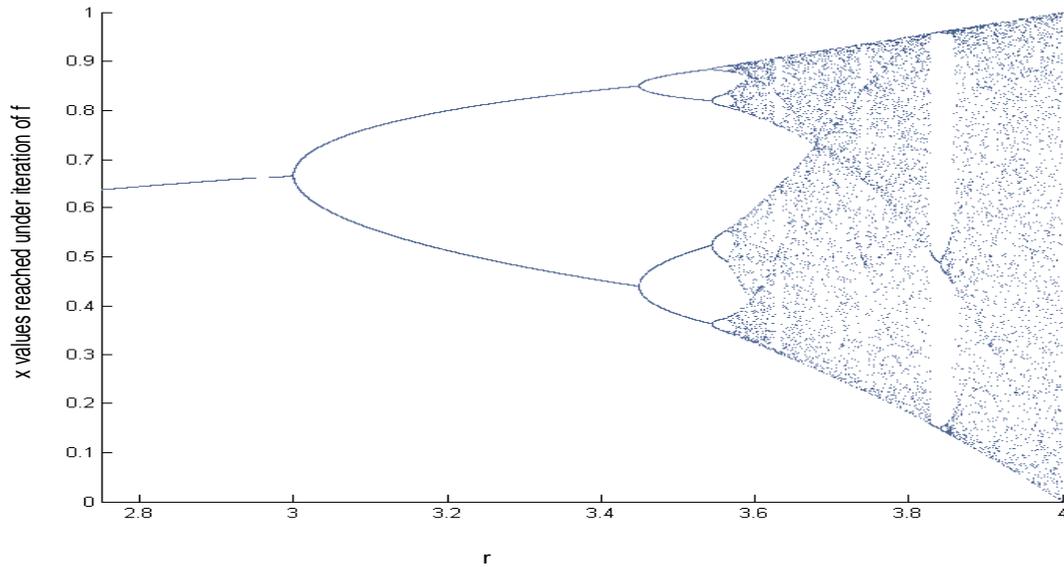


Figure 2.4: Bifurcation diagram for the logistic map.

To understand the behavior of this map better, we consider the fixed points of the second iterate of f , which can be thought of as either p and q such that $f(p) = q$ and $f(q) = p$ or the two solutions of $f^2(x) = x$. This is a quartic polynomial, which can be simplified by noting that the origin and $x^* = 1 - \frac{1}{r}$ are both solutions, (since in both cases, $f(x^*) = x^*$) and therefore $f^2(x^*) = x^*$) we factor these solutions out of the polynomial and with a little algebra find that

$$p, q = \frac{r + 1 \pm \sqrt{(r-3)(r+1)}}{2r}.$$

From this we can see that for $r > 3$ the solutions are real and so, for any $r > 3$ the logistic map has a 2-cycle. Now we will consider stability. The 2-cycle is stable when p and q are stable fixed points, that is, when $|\frac{d}{dx}f^2(p)| = |f'(f(p))f'(p)| = |f'(q)f'(p)| < 1$ which by symmetry is the same for q . Plugging in for p and q this simplifies to the condition

$|-r^2 + 2r + 4| < 1$ or $3 < r < 1 + \sqrt{6}$ (taking into account the domain of r). For values of $r > 1 + \sqrt{6}$ the 2-cycle is unstable, and therefore does not appear in the bifurcation diagram, however, these unstable areas play an important role in the dynamics of the map. In fact, at $r = 1 + \sqrt{6}$ there is a saddle node bifurcation to a 4-cycle [108]. This bifurcation pattern will continue,

producing a period-doubling cascade, which is one of the most well known routes to chaos [27]. The logistic map displays rich self-similar behavior and a complicated intermingling of period-doubling cascades and chaos [102].

Denote r_n as the r value for which the 2^n -cycle first appears. From the bifurcation diagram we can only discern the first few locations of period-doubling, however, there are infinitely many. The sequence $\{r_n\}$ is an infinite series called a period doubling cascade, where a 2^n -cycle exist for every positive integer n . Successive period doubling bifurcations occur closer and closer together. The location converges geometrically to the accumulation point $r_\infty = 3.569946\dots$. It was discovered by Feigenbaum [38] that the distance between successive bifurcations shrinks by a constant factor:

$$\delta = \lim_{n \rightarrow \infty} \frac{r_n - r_{n-1}}{r_{n+1} - r_n} = 4.6692016291\dots$$

known as the Feigenbaum constant. δ is a universal constant for the rate of convergence of bifurcation locations for maps approaching chaos through period doubling. This is in fact a very large class of systems!

Since, from the perspective of persistent homology, all necessary information about a function is encoded in the critical values, it is possible to compute the persistent homology of the orbit of a one dimensional discrete time dynamical system.

2.1.2 Pattern in the Persistence Diagram

Persistent homology is a novel way to approach these systems. When persistence is computed in this way, there is a distinct pattern that appears in the persistence diagram. All the persistence points in the persistence diagram fall along a curve. We will restrict ourselves to considering $r < 3$, since the dynamics for $r < 3$ are not particularly interesting.

We will start in the simplest case, let all the points in our orbit be members of a 2-cycle, $\{x_1, x_2\}$. Without loss of generality assume that $x_1 < x_2$. This means that x_1 will be a local minimum, and x_2 will be a local maximum. If we compute persistence on this data, we will end

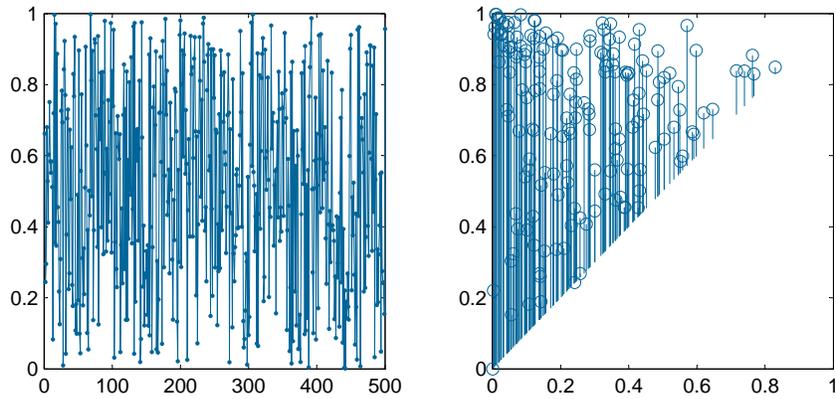


Figure 2.5: Points drawn from uniform distribution and the associated persistence diagram. There is no clear pattern in the way that the persistence points are arranged.

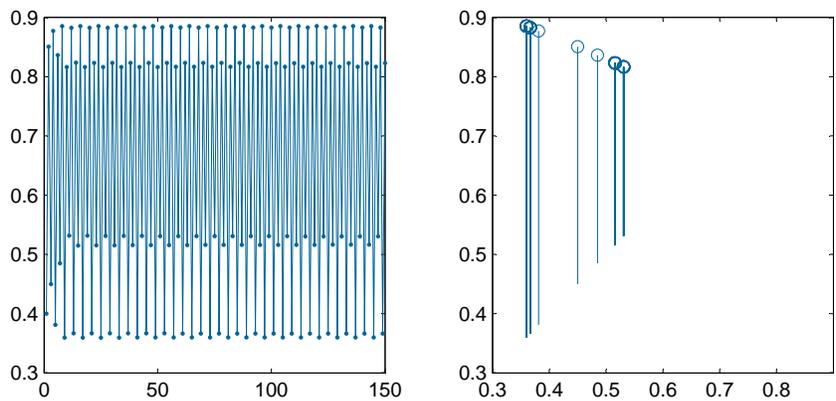


Figure 2.6: An orbit of the logistic map for parameter values $r = 3.545$, which produces a stable 8-cycle. There are four bars with high multiplicity associated to the 8-cycle. Even the bars associated with the orbit prior to stabilizing fall along the same line.

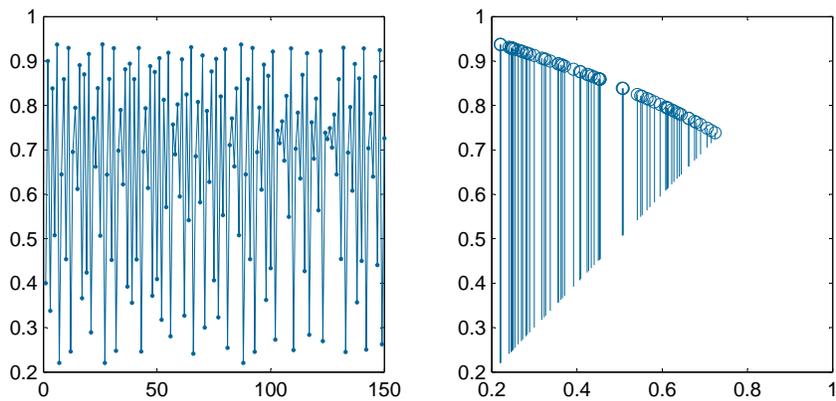


Figure 2.7: An orbit of the logistic map for parameter value $r = 3.75$, which is in the chaotic regime. The points in the persistence diagram all fall along the same curve.

up with the persistence point (in birth-death coordinates) (x_1, x_2) with multiplicity up to the number of cycles that are present in our data.

Most persistence points (for large enough orbits) fall on this expected curve. In fact, for many choices of initial conditions, this is the only pattern appearing. Occasionally, a few points in the persistence diagram may break the dominant curve pattern in the persistence diagram. This is caused when a few of the initial points break the dominant pairing pattern (which will be discussed in detail later). We will address “edge effects” truncating the first several points up until a maximum. These effects disappear with this caveat.

One might expect a clear pattern in the persistence diagram of a periodic orbit with a higher period. For example, a 16-cycle, the persistence diagram will have eight persistence points with multiplicity, since for the logistic map, if there will be four minima and four maxima in the cycle. The remarkable feature here is that a clear pattern emerges in the locations of the persistence points for the behavior leading up to a stable periodic cycle and even more remarkably, a clear pattern emerges for the chaotic regime as well. The persistence points in fact fall on a curve for the logistic map. To seek to understand this pattern in the persistence diagram, we will first investigate a simpler map with similar dynamics, namely the tent map.

2.1.3 The Tent Map

The tent map is a one dimensional discrete system given by

$$x_{n+1} = \mu \min(x_n, 1 - x_n)$$

with $1 \leq \mu \leq 2$, mapping the unit interval to itself. This will provide insight because the tent map and logistic maps are topologically conjugate, but the tent map is computationally more accessible. Topological conjugacy preserves topological features like periodicity.

Consider two metric spaces X, Y and continuous maps $f : X \rightarrow X$ and $g : X \rightarrow Y$, f and g are said to be topologically conjugate if there exists a homeomorphism $h : X \rightarrow Y$ such that $h \circ f(x) = g \circ h(x)$ for every $x \in X$. [4] The homeomorphism is given by $h(x) = \sin^2\left(\frac{\pi}{2}x\right)$ So that if

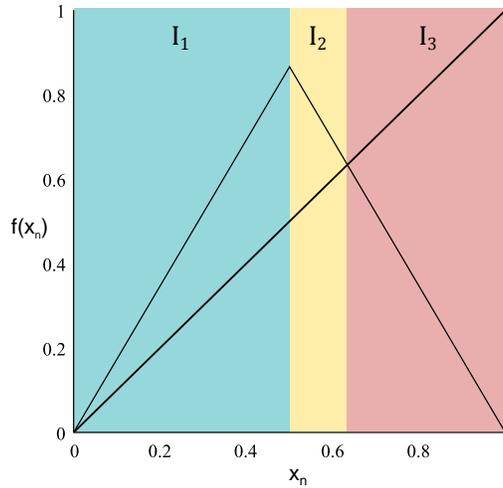


Figure 2.8: The updating function for the tent map is plotted above. The three intervals of the domain I_1 , I_2 , and I_3 will play a role in understanding the dynamics and ultimately the pattern in the persistence diagram.

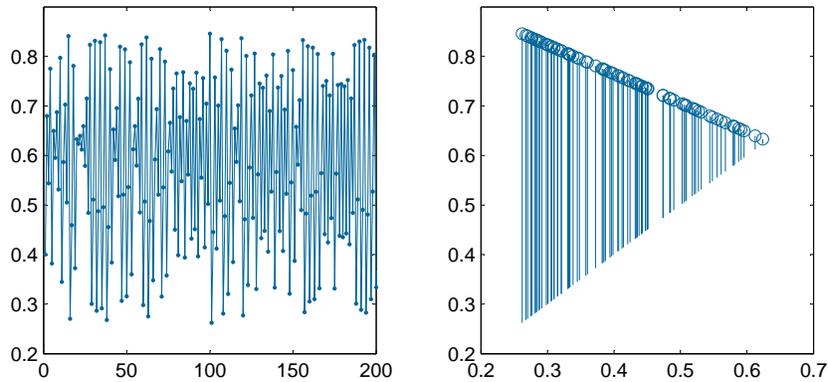


Figure 2.9: Iterations of the tent map and the associated persistence diagram.

$T(x)$ is the tent map and $L(x)$ is the logistic map, $L(x) = h \circ T \circ h^{-1}(x)$ [75]. This means that these maps behave the same dynamically [99]. There is a one-to-one correspondence between periodic orbits of topologically conjugate maps. Quantitatively these maps may be different, but qualitatively the two maps display the same behavior.

Stable orbits undergo a period doubling bifurcation that, like the logistic map, cascade to chaos. At any given parameter for the logistic map, the tent map at the conjugate parameter has the same number of unstable periodic orbits [88]. Further, Feigenbaum's scaling by δ and α are present. As seen in Figure 2.9, there is a clear, decreasing line in the persistence diagram

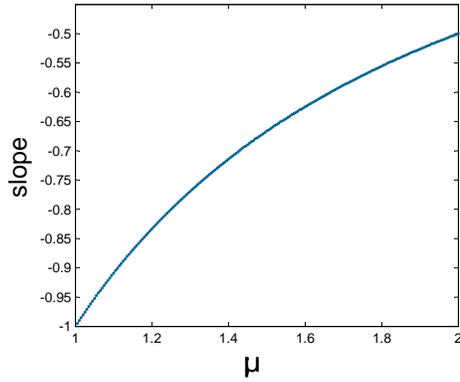


Figure 2.10: The persistence points for the tent map will fall along a line with slope $\frac{-1}{\mu}$. This means that computing the slope of the points in the persistence diagram will recover the parameter μ driving the dynamics.

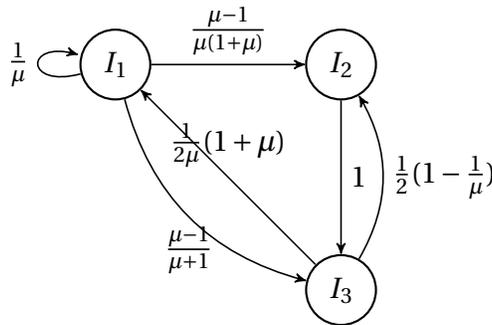


Figure 2.11: The Markov graph for the tent map. The intervals are shown in Figure 2.8, each edge gives the probability that a point will move from one interval to the next under a single iteration of the map.

both in the periodic and chaotic regime. We will justify why this pattern appear and connect these ideas to some tools from symbolic dynamics.

Consider a collection of closed subintervals $\{I_k\}$ of the interval I . These intervals form a partition if the interior of the intervals are pairwise disjoint [23]. A *Markov graph* of $f : I \rightarrow I$ associated with the partition I_k is the graph whose vertices are the intervals of the partition and the edges of pairs (I_i, I_j) such that $f(I_i) \supset I_j$. Such an edge is denoted $I_i \rightarrow I_j$. Every path in the Markov graph is associated with a point whose itinerary follows the path. If the path is closed, then there exists a periodic point (with period equal to the length of the path). Note the intervals of the partition are not disjoint (only their interiors are) so there may exist points with period smaller than the length of the cyclic path.

When determining the intervals for the Markov graph, we first divide phase space into intervals that are orientation preserving and orientation reversing. Orientation preserving means that for $x < y$, $f(x) < f(y)$. Similarly, orientation reversing means that $x < y$, $f(x) > f(y)$. Next break these down to intervals where $x_{n+1} = f(x_n) > x_n$ (the orbit is increasing) or where $x_{n+1} = f(x_n) < x_n$ (the orbit is decreasing). The resulting intervals are illustrated in Figure 2.8. If a point falls in the interval $I_1 = [0, 0.5]$, then applying the tent map to x_n will give $\mu x_n \geq x_n$ since $1 \leq \mu \leq 2$. Similarly, for points $x_n \in I_2$ where $I_2 = [0.5, \frac{\mu}{1+\mu}]$ under the tent map will map to $\frac{\mu}{1+\mu} < \mu(1 - x_n) < 0.5\mu$ so the next iteration: $\mu(1 - x_n) > x_n$ since $x_n < \frac{\mu}{\mu+1}$. Therefore, for almost every point in the interval $I_1 \cup I_2$ the orbit is increasing ($f(x_n) > x_n$). So minima may occur in $I_1 \cup I_2$, but maxima will not. For $x_n \in I_3$ where $I_3 = [\frac{\mu}{\mu+1}, 1]$ applying the tent map give rise to $\mu(1 - x_n) < \frac{\mu\mu}{\mu+1}$. Therefore points in this region are decreasing. Maxima will occur in I_3 , but minima will not.

Claim. *If $x_n \in I_1 \cup I_2$, then $f^{-1}(x_n) \in I_3$.*

Stated in another way, any minimum is computed directly from a point in I_3 . Since the orbits of all points in $I_1 \cup I_2$ are increasing, if a point was computed from a point in that interval, then it necessarily came from a smaller point in the same interval, which means that the point itself cannot be a minimum. So minima are computed from I_3 .

Now consider how the level set persistence diagram is built. First, the only points that will contribute are maxima and minima. Any intermediate point is disregarded. Minima are associated with the birth of a new feature and maxima are associated with the death of a feature. As features (or connected components of the level set) are born and die, this information is recorded in a persistence diagram. The function value at the birth of a feature is on the horizontal axis and the function value of the death of a feature is on the vertical axis. A bar is then drawn from the diagonal to this point. The persistence diagram captures information regarding the size of geometric features. A long bar signifies a significant feature. Consider a level set rising from below the graph of the function. As it encounters a minimum, it splits into two connected components, and this is the birth of a feature. The level set continues to rise and divide as it

reaches a new minimum. When a feature encounters a maximum, there are two minima associated with the level set (one on the left and one on the right). Since these are the associated births of the feature, they are the candidates for pairing with the maximum to give a description of the longevity of the feature. The largest minimum is taken to be descriptive of the feature that disappears at this maximum, and is thus these two values are paired into the persistence. This pairing scheme along with the structure of the tent map leads to a line in the persistence diagram for the tent map.

The following argument gives a justification for this pattern. Consider the smallest maximum point. We have already seen that there is a division between minima and maxima at $\frac{\mu}{\mu+1}$. So this maximum is the smallest point above $\frac{\mu}{\mu+1}$.

Claim. *The iteration immediately following the smallest maxima is the largest minima and will pair with the smallest maxima.*

Suppose, by way of contradiction, that $f(M_0) = m_0$ is not the largest minimum. So there exists m_n such that $\frac{\mu}{1+\mu} > m_n > m_0$. The first inequality is enforced since the point must be increasing in the next iteration to be a minimum. m_n is an iteration of the tent map, so $m_n = f(M_n)$, where M_n is the previous iteration. M_n cannot be on the first branch, since that would mean $M_n < m_n$, but m_n is a minima. This means that $m_n = \mu(1 - M_n)$. Similarly, $m_0 = \mu(1 - M_0)$. Substituting into $m_n > m_0$ and using the fact that all iterations are in the interval $[0, 1]$, this implies $M_n < M_0$, but by assumption, M_0 is the lowest maximum, which is a contradiction.

This means that the lowest maximum (which is the first to pair) will pair with the largest minimum, which is in fact the next iteration in the tent map. In general, iterating the tent map gives rise to a distance stretching. By the same reasoning as above, the higher the maximum is, the lower the subsequent minimum will be. Minima that have a greater value are come from a maximum with a lower value. This creates a “fanning out” pattern. Eventually, a maximum will be large enough that it will map to a small minimum that will need to “climb up” again before re-entering the cycle of a maximum mapping to a minimum. Therefore a point in the persistence

diagram, which is given as (birth, death) can be written as $(f(x_n), x_n) = (\mu(1 - x_n), x_n)$ where x_n is a maximum.

Claim. *The points in the persistence diagram for the tent map fall along the curve*

$$d = 1 - \frac{b}{\mu},$$

where b is the birth coordinate and d is the death coordinate in the persistence diagram.

Following the same reasoning, for similarly defined intervals:

$$I_1 = [0, \bar{x}], I_2 = [\bar{x}, x^*] I_3 = [x^*, 1]$$

where $\bar{x} = 0.5$ is the location of the maximum of the updating function and x^* is the nonzero fixed point, we can make the following claim for the logistic map.

Claim. *The points in the persistence diagram for the logistic map falls along the curve*

$$d = \frac{\sqrt{r} + \sqrt{r - 4b}}{2\sqrt{r}},$$

where b is the birth coordinate and d is the death coordinate in the persistence diagram.

In the next section we will show where this pattern comes from for a large class of maps, of which the logistic map is an example.

2.2 Persistence of Unimodal Maps

Unimodal maps have a lot of structure [29, 86]. Such maps were studied extensively by Feigenbaum and admit the same universality constants δ and α [39]. The logistic map is a unimodal map. So is the quadratic family of maps $f(x_{n+1}) = c - x_n^2$. The bifurcation diagram of the family of quadratic maps is not only qualitatively ubiquitous, but quantitatively as well [23]. The existence of a period-doubling cascade in a unimodal map guarantees the existence of

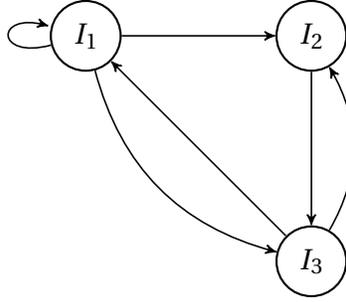


Figure 2.12: The Markov graph for a unimodal map restricted to the dynamic core.

chaotic behavior [69]. The chaotic regime is densely interspersed in the periodic regime [111]. The tent map is not technically considered a unimodal map because the maximum is not differentiable, however since this is a single point (a set of measure zero) it does not effect the dynamics [15].

This can be generalized to a class of one-dimensional difference equations called unimodal maps, which we define as follows. Given a map $f : I \rightarrow I$, where I is the unit interval and f is continuous with a unique maximum at $c \in I$, and increases to c and decreases away from c . Without loss of generality, let $f(0) = 0$ and $f(1) = 0$. There are two equilibria, 0 and x^* . We also insist that $c < x^*$. Define $I_1 = [0, c]$, the region on which f is increasing. $I_2 = [c, x^*]$, which is the region on which f is decreasing, to the left of the equilibrium point x^* . $I_3 = [x^*, 1]$, is the region to the right of the equilibrium point, f is decreasing on I_0 as well.

For f and intervals defined in that way, we can build a Markov graph, pictured in Figure 2.12. $f(I_2) = [f(x^*), f(c)] = [x^*, f(c)] = I_3$ since f is continuous, decreasing, and therefore orientation reversing. $f(I_3) = [f(x^*), f(1)] = [0, x^*] \subseteq I_1 \cup I_2$. It is possible for I_3 to map into the entire interval $I_1 \cup I_2$.

Recall, for discrete maps, we consider a point in the orbit, x_k , to be a local minimum if $x_{k-1} > x_k$ and $x_{k+1} > x_k$. Likewise, if $x_{k-1} < x_k$ and $x_{k+1} < x_k$, then we consider x_k to be a local maximum. Consider a finite subset of an orbit of f : $x_0, x_1, x_2, \dots, x_n$.

Claim. For any $x_k \in I_1 \cup I_2$, x_k is not a maximum.

Proof. Since $x_0^* < c < x^*$, f is continuous, increasing to c , decreasing away from c and $f(x_0^*) = x_0^*$, $f(x^*) = x^*$, the orbit of f is increasing. So, for $x_k \in I_1 \cup I_2$ $f(x_k) \geq x_k$, which means that x_k is not a maximum. \square

Claim. For any $x_k \in I_3$, x_k is a maximum.

Proof. $x_k \in I_3$ implies that $f^{-1}(x_k) \subseteq I_1 \cup I_2$, by the definition of the intervals, $\max(f^{-1}(x_k)) < x_k$. Further, $f(x_k) \in I_1$ implies that $f(x_k) < x_k$ by the definition of the intervals. \square

Claim. x_k is a minimum if and only if $f^{-1}(x_k) = x_{k-1} \in I_3$.

Proof. \Rightarrow) If x_k is a minimum, by the previous claim, it cannot be in I_3 , so $x_k \in I_1 \cup I_2$. If $x_k \in I_1$ then by definition of the intervals, $f^{-1}(x_k) \in I_1 \cup I_3$. If $f^{-1}(x_k) \in I_1$, then $f^{-1}(x_k) < x_k$ and x_k is not a minimum. So $f^{-1}(x_k) \in I_3$.

Similarly, if $x_k \in I_2$, then by definition of the intervals, $f^{-1}(x_k) \in I_1 \cup I_3$. If $f^{-1}(x_k) \in I_1$, then $f^{-1}(x_k) < x_k$ and x_k is not a minimum. So $f^{-1}(x_k) \in I_3$.

\Leftarrow) If $f^{-1}(x_k) = x_{k-1} \in I_3$, by definition of the intervals, $f^{-1}(x_k) > x_k$. Further, $x_k \in I_1 \cup I_2$. On $I_1 \cup I_2$, the orbit of f is increasing, so $f(x_k) > x_k$, which means that x_k is a minimum. \square

Claim. For a finite subset of the orbit of a point under the map f , let x_M be the smallest maximum, then $x_{M+1} = f(x_M)$ is the largest minimum.

Proof. Since x_M is a maximum, by claim 2, $x_M \in I_3$. On I_3 , f is continuous and decreasing, so $f(\min\{x_k | x_k \in I_3\}) = \max\{f(x_k) | x_k \in I_3\}$ \square

Claim. The persistence points of a unimodal map defined on (a, b) and stable equilibrium x^* will have the form $(f(x_n), x_n)$, where $x_n \in (x^*, b)$

Proof. When building the persistence diagram, we only consider points that are maxima or minima. By Claim 6, all maxima will be in I_3 . \square

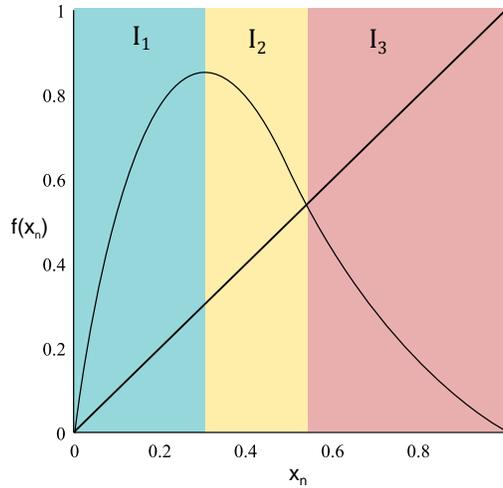


Figure 2.13: A general unimodal map with the intervals marked.

2.2.1 Examples of Unimodal Maps

There are many examples of unimodal maps, several examples follow. One family of maps has the form

$$f(x_{n+1}) = 1 - a|x_n|^k$$

for $a \in (0, 2]$. When $k = 1$ this is equivalent (under a change of variables) to the tent map and when $k = 2$ this is equivalent to the Quadratic family of maps [8]. This map is unimodal according to the definition in the previous section under the transformation that brings $f(0) = f(1) = 0$. An example of iterations and the associated persistence diagram is shown in Fig. .

Two other families of unimodal maps are the Gauss map, given by

$$f(x_{n+1}) = e^{-ax_n^2} + b$$

and the sine map, given by

$$f(x_{n+1}) = \lambda \sin(x_n)$$

Both of which give rise to similar patterns in the associated persistence diagrams.

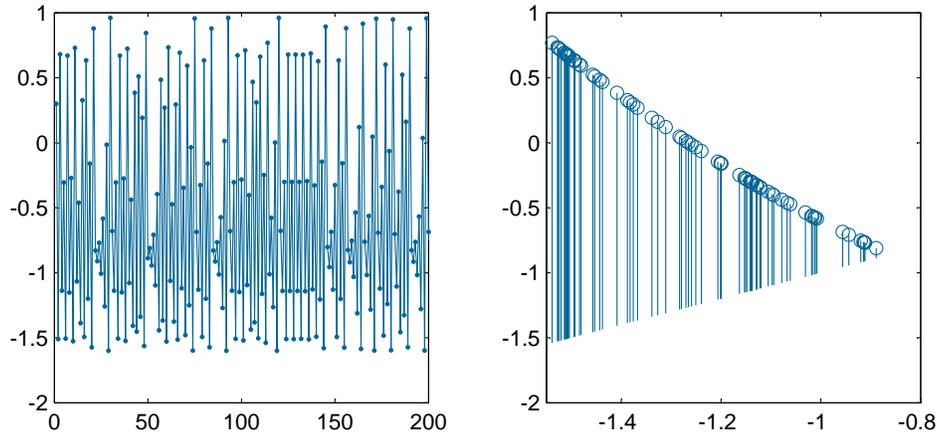


Figure 2.14: Iterates and the associated persistence diagram of the Quadratic Map for $a = 1.6$.

2.3 Poincaré Maps of Higher Dimensional Systems

We see this pattern emerging in the persistence of one-dimensional sections of higher dimensional differential equations. The essential behavior of higher dimension dynamical systems can often be reduced to a one-dimensional map [23]. This can be done by looking at cross-sections of trajectories of an orbit. This is called the first-return map or the Poincaré map and represents the system by a discrete difference equation. There are some natural conditions to the choice of cross-sections [88].

Unimodal maps are two-to-one, so we would not expect them to show up as the Poincaré map of a two dimensional system, however, they do appear as Poincaré maps of three-dimensional systems [36]. In fact, it was through looking at a one-dimensional map of the Lorenz attractor that Lorenz his model for atmospheric advection [72]. Lorenz was able to construct an approximate one dimensional map from this system by plotting successive maximums [36, 72]. When this is done for the z variable, this gives rise to a single peaked map. Lorenz was able to use this to convincingly argue that the system was chaotic [75]. We can detect by computing persistence on the projection of the trajectory in each dimension. The Rössler attractor also has unimodal maps as the backbone. Because the attractor is so compressive, for all practical purposes, the return map of the Rössler system is one dimensional.

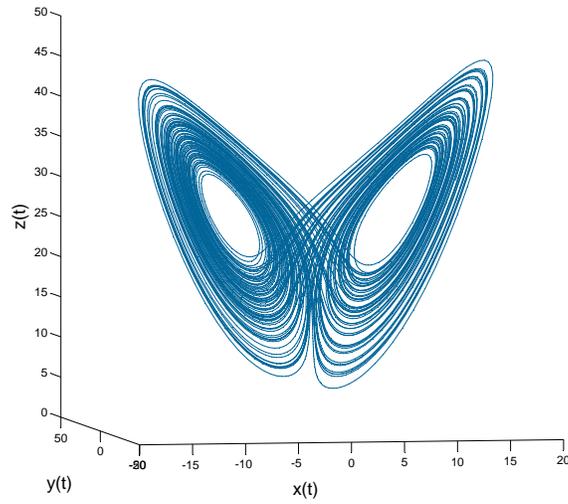


Figure 2.15: Example of a trajectory of the Lorenz attractor.

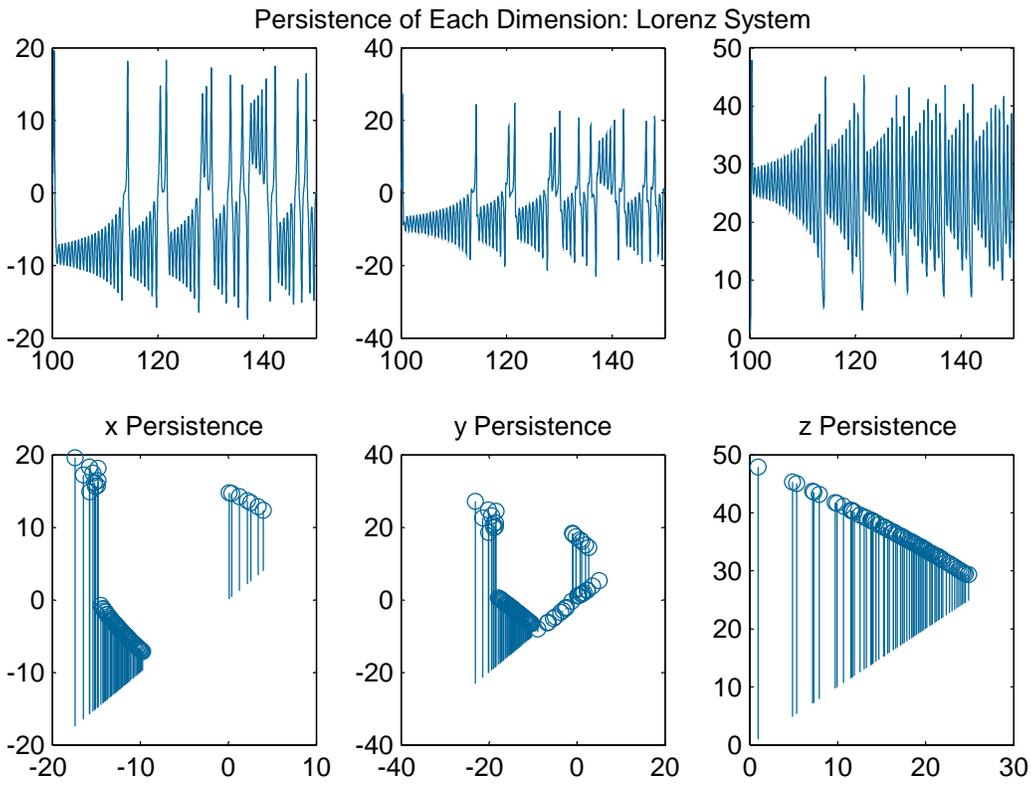


Figure 2.16: (Top) One-dimensional maps in each dimension (x,y, and z) in time generated by projecting the orbit onto the x,y, and z coordinates respectively. The persistence computed for each one-dimensional time series (Bottom).

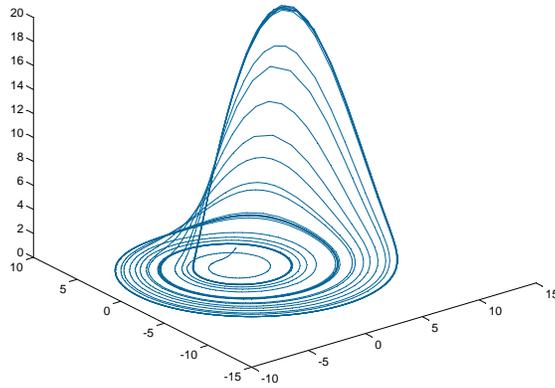


Figure 2.17: Example of a trajectory of the Rössler attractor.

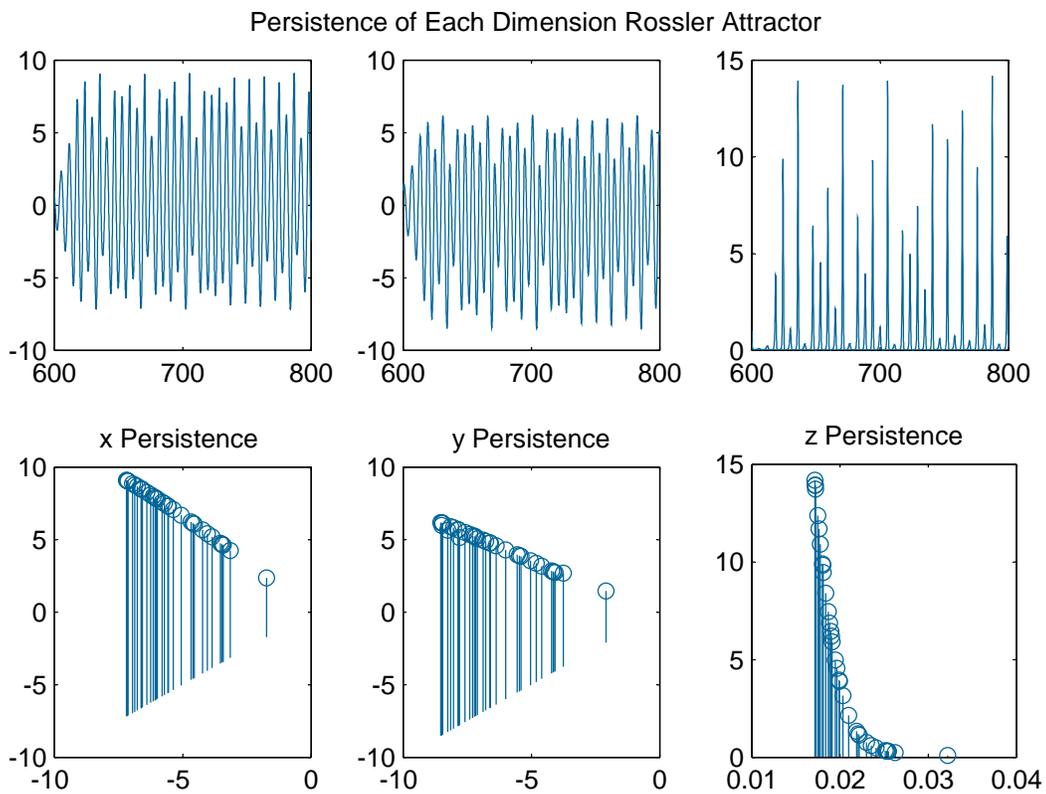


Figure 2.18: (Top) One-dimensional maps in each dimension (x,y, and z) in time generated by projecting the orbit onto the x,y, and z coordinates receptively. The persistence computed for each one-dimensional time series (Bottom).

2.3.1 Noisy Maps

The points on the persistence diagram for the logistic map fall along the line given by

$$d = \frac{1}{2} + \frac{1}{2} \sqrt{1 - \frac{4d}{r}}$$

where (b, d) gives a point in the persistence diagram in birth-death coordinates. We can use this not only to fit data for an unknown r value, but also to fit noisy data.

We generate noisy data from the logistic map in several controlled ways. In the first, we perturb each iteration by a small amount before the map evolves again. This is given by the following dynamical system:

$$x_{n+1} = rx(1 - x) + \sigma v$$

where σ is a scale parameter on the amount of noise and v is a Gaussian random number. In this case, it is possible, but rare, for a point to be mapped outside of the unit interval (especially for $r = 4$). If this occurs, we would disregard that orbit. However, this event was rare enough that it was not necessary in our computations.

We generated 1000 examples for each parameter value and noise level. Each example has a random initial condition and 1000 iterations. We compute persistence of each orbit and use a nonlinear least squares algorithm to fit to the expected function of the persistence points for the logistic map. We take the mean of the difference between r_{fit} and r . The results are shown in Table 2.2. Table 2.3 shows the standard deviation of the set of all r_{fit} for each parameter and noise level pair. For noisy data generated in this way, in almost every case, r_{fit} is slightly higher than r , but is very close to the actual value that was used to generate the data. We also notice that there is very small variance of the fit parameter across all trials which indicates that this is a promising method for parameter recovery, even in the presence of noise.

Table 2.2 and Table 2.5 show similar results for data generated with a noisy value for r .

$$r_n = \min(\max(r + \sigma v, 3)4)$$

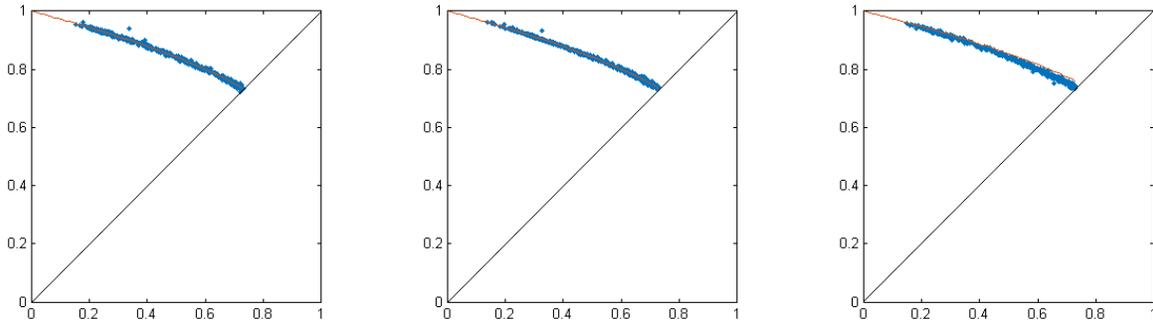


Figure 2.19: (Left) The persistence diagram and the fit for data from the logistic map with a noisy value for the parameter r . Here $r = 3.75$ and $r_{fit} = 3.7508$ and $\sigma = 0.05$. (Center) The persistence diagram and the fit for data from the logistic map with a noisy value for the parameter r . Here $r = 3.75$ and $r_{fit} = 3.7503$ and $\sigma_x = 0.01$ and $\sigma_r = 0.01$ (Right) The persistence diagram and the fit for data from the logistic map with each iteration perturbed by a small random amount. Here $r = 3.75$ and $r_{fit} = 3.7498$ and $\sigma = 0.01$

Table 2.2: This table gives the difference between the mean r_{fit} and the actual parameter r for data from the logistic map with each iteration perturbed by a small random amount.

σ	$r = 3.6$	$r = 3.65$	$r = 3.7$	$r = 3.75$	$r = 3.8$	$r = 3.85$	$r = 3.9$	$r = 3.95$	$r = 4$
0.001	0.0026	0.0022	0.0017	0.0029	0.0036	0.0203	0.0072	0.0068	0.0098
0.005	0.0032	0.0023	0.0019	0.0029	0.0044	0.0066	0.0067	0.0073	0.0097
0.01	0.0041	0.0027	0.0024	0.0031	0.0045	0.0066	0.0068	0.0084	0.0094
0.05	0.0150	0.0180	0.0171	0.0173	0.0187	0.0196	0.0206	0.0236	0.0249
0.1	-0.0044	0.0031	0.0098	0.0179	0.0254	0.0302	0.0361	0.0430	0.0509

where σ is a controlled scale parameter for v a random Gaussian value. r_n is the value of r used to generate x_{n+1} . We enforce that $3 \leq r \leq 4$ so that we do not slip into the regime where 0 is an attracting fixed point. The final way that noise is incorporated is in both the value of r and each iteration. Even in this case, the parameter fit is still close to the actual parameter value. This pattern in the persistence diagram appears to be robust to noise that is introduced in the system in various ways. Our findings for the noisy trails are summarized in the following tables.

Another method one might consider for parameter discovery is fitting the return map. This method involves plotting the points $(x_n, x_{n+1}), (x_{n+1}, x_{n+2})$ and so on. This technique embeds the one-dimensional map into 2-dimensions. One would then fit the resulting points to the known map. This is effective for detecting deterministic behavior [72]. However, this method is very sensitive to a single missing point and does not provide as good of a fit.

Table 2.3: This table gives the standard deviation of r_{fit} and the actual parameter r for data from the logistic map with a small amount of noise added at each iteration.

σ	$r = 3.6$	$r = 3.65$	$r = 3.7$	$r = 3.75$	$r = 3.8$	$r = 3.85$	$r = 3.9$	$r = 3.95$	$r = 4$
0.001	0.0034	0.0026	0.0021	0.0034	0.0040	0.0236	0.0075	0.0071	0.0093
0.005	0.0035	0.0029	0.0029	0.0038	0.0048	0.0073	0.0070	0.0073	0.0094
0.01	0.0044	0.0042	0.0044	0.0049	0.0061	0.0079	0.0083	0.0092	0.0108
0.05	0.0176	0.0190	0.0200	0.0207	0.0212	0.0235	0.0242	0.0278	0.0263
0.1	0.0351	0.0369	0.0363	0.0384	0.0403	0.0395	0.0422	0.0441	0.0461

Table 2.4: This table gives the difference between the mean r_{fit} and the actual parameter r for data generated with a noisy value for r .

σ	$r = 3.6$	$r = 3.65$	$r = 3.7$	$r = 3.75$	$r = 3.8$	$r = 3.85$	$r = 3.9$	$r = 3.95$	$r = 4$
0.001	0.0024	0.0022	0.0017	0.0030	0.0037	0.0779	0.0071	0.0072	0.0093
0.005	0.0025	0.0020	0.0017	0.0029	0.0035	0.0164	0.0071	0.0070	0.0078
0.01	0.0025	0.0021	0.0017	0.0030	0.0038	0.0092	0.0068	0.0068	0.0058
0.05	0.0024	0.0017	0.0020	0.0022	0.0034	0.0046	0.0053	0.0034	-0.0115
0.1	0.0019	0.0018	0.0015	0.0011	0.0017	0.0003	-0.0039	-0.0136	-0.0326

Table 2.5: This table gives the standard deviation of r and the actual parameter r for data generated with a noisy value for r .

σ	$r = 3.6$	$r = 3.65$	$r = 3.7$	$r = 3.75$	$r = 3.8$	$r = 3.85$	$r = 3.9$	$r = 3.95$	$r = 4$
0.001	0.0033	0.0026	0.0020	0.0035	0.0041	0.0817	0.0073	0.0070	0.0093
0.005	0.0034	0.0026	0.0021	0.0033	0.0039	0.0177	0.0072	0.0070	0.0094
0.01	0.0032	0.0025	0.0022	0.0034	0.0042	0.0098	0.0071	0.0067	0.0093
0.05	0.0039	0.0038	0.0042	0.0047	0.0057	0.0069	0.0078	0.0086	0.0090
0.1	0.0065	0.0065	0.0071	0.0076	0.0082	0.0090	0.0094	0.0094	0.0097

Table 2.6: This table gives the difference between the mean r_{fit} and the actual parameter r for data generated with a noisy value for r and with noise added to each iteration. We choose to keep both noise factors the same, $\sigma_x = \sigma_r = \sigma$.

σ	$r = 3.6$	$r = 3.65$	$r = 3.7$	$r = 3.75$	$r = 3.8$	$r = 3.85$	$r = 3.9$	$r = 3.95$	$r = 4$
0.001	0.0025	0.0021	0.0017	0.0030	0.0036	0.0201	0.0076	0.0070	0.0094
0.005	0.0030	0.0022	0.0018	0.0029	0.0041	0.0072	0.0070	0.0072	0.0072
0.01	0.0039	0.0025	0.0023	0.0029	0.0044	0.0065	0.0073	0.0081	0.0055
0.05	0.0151	0.0159	0.0167	0.0177	0.0184	0.0196	0.0217	0.0203	0.0037
0.1	-0.0089	-0.0011	0.0038	0.0151	0.0179	0.0235	0.0228	0.0208	0.0056

Table 2.7: This table gives the standard deviation of r and the actual parameter r for data generated with a noisy value for r and with noise added to each iteration. We choose to keep both noise factors the same, $\sigma_x = \sigma_r = \sigma$.

σ	$r = 3.6$	$r = 3.65$	$r = 3.7$	$r = 3.75$	$r = 3.8$	$r = 3.85$	$r = 3.9$	$r = 3.95$	$r = 4$
0.001	0.0033	0.0026	0.0021	0.0034	0.0040	0.0222	0.0076	0.0071	0.0093
0.005	0.0036	0.0029	0.0029	0.0038	0.0048	0.0076	0.0073	0.0074	0.0099
0.01	0.0044	0.0042	0.0044	0.0050	0.0061	0.0076	0.0086	0.0091	0.0106
0.05	0.0184	0.0186	0.0201	0.0207	0.0222	0.0239	0.0260	0.0264	0.0268
0.1	0.0355	0.0383	0.0351	0.0387	0.0395	0.0430	0.0423	0.0439	0.0456

Chapter 3

Parameter Classification of Pattern-Forming Systems

3.1 Introduction

Complex patterns in nature are ubiquitous and can be observed in systems of vastly different scales; from the distribution of matter in the universe [18], to near-periodic ripples in sand dunes or stripes on a zebra down to nanoscale structures formed by ion bombardment [6]. Pattern-formation is a prominently studied area of nonlinear dynamics and understanding such systems is important to a wide variety of fields in the scientific community such as biology, physics, engineering, and chemistry.

Often these patterns occur in nonlinear systems that are driven from equilibrium, displaying spatial and temporal variations. There has been a lot of work done recently [95] in the study of nonlinear systems driven from equilibrium by, for example, a gradient in temperature, concentration or velocity [109]. These systems are challenging due to a lack of established theoretical frameworks [59].

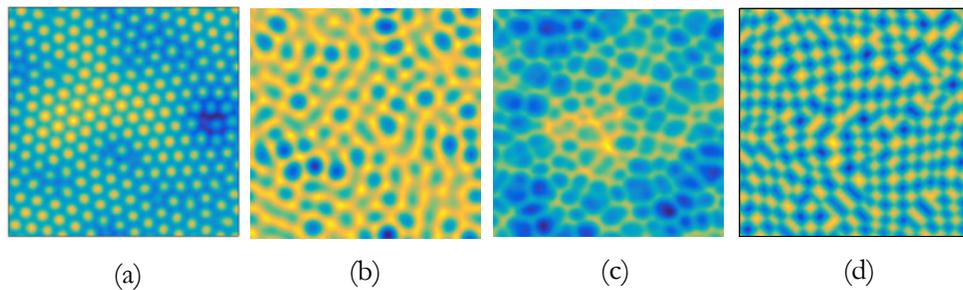


Figure 3.1: Examples of pattern formation that will be discussed in this chapter. (a) Model of the surface of a binary compound bombarded with a normal-incidence ion beam.(Section 5.4) (b) and (c) and early and late time examples of a system that models, for example solidification of a melt (Section 3.4) (d) Model of pyramidal structures formed on irradiated Geranium (Section 4.3)

Self-organized pattern formation is frequently the result of interactions a number of different phenomenological processes. On a coarse scale, we may understand mechanisms that contribute to pattern formation in a number of systems. However, many of the particular details or complex interactions of physical mechanisms are not fully understood [18, 95]. It can often be the case that these poorly-resolved or poorly understood processes are parametrized rather than treated explicitly. Because of this, it becomes important to determine the influence of model parameters on the system. There are a variety of methods to do this, many of which require computationally expensive simulations [2]. Irregular time-varying structures and complexity of patterns, and sensitivity to initial conditions, among other things, makes quantifying or even distinguishing patterns difficult [28]. Recently there has been much interest in using topological methods in pattern formation and pattern evolution, in particular in material sciences [118]. Computational topology has emerged as a tool that retains essential information for studying patterns, but significantly reduces the dimensionality of the data [22]. For example, persistent homology has been used to distinguish between parameters for complex patterns formed through a phase separation process [28].

We begin this section with a brief introduction to pattern formation in Section 3.2. We will use persistence images, introduced in 1.6, which are a vector representation of persistence diagrams that allow for the use of machine learning techniques [2]. We apply these techniques to pattern-forming systems. The first is from the anisotropic Kuramoto–Sivashinsky equation as a way to distinguish by anisotropy parameter (Section 3.4). The next models the formation of pyramidal and inverted pyramidal structures on an irradiated surface (Section 4.3). Persistent homology provides a valuable tool for distinguishing parameters in these complex spatio-temporal patterns.

3.2 Pattern Formation

We give a brief introduction to pattern formation in systems driven from equilibrium. Given a system of partial differential equations (PDEs) $\partial_t U = F(U, \partial_x U, \dots, R)$ where R is a control pa-

parameter, x is a d -dimensional variable and $U = u_1(x, t), \dots, u_n(x, t)$ and boundary conditions and initial conditions, we will investigate the response of the system to single Fourier mode disturbances of the system. Linearizing $F[U]$ around a base state, that is, a solution to the equation of motion near equilibrium, will reveal basic instabilities. [109]

To understand these stabilities, study the evolution of modes of a given wave vector k in $u_j(x, t) = u_0 e^{ikx + \sigma t}$ [19]. The resulting linear equations will give rise to eigenvalues $\sigma_\alpha(k)$. The one with the largest real part is denoted $\sigma(k)$. $Re(\sigma(k))$ is the growth rate. Interestingly, for most pattern forming systems, k is in a space of dimension 1 or 2, so the pattern formation largely occurs on a surface [19]. Assume that for the control parameter R , there is a critical point R_c at which $Re(\sigma(k_0)) = 0$ at some wave vector k_0 , and for parameter values $R < R_c$, $Re(\sigma(k)) < 0$. This means that a small perturbation of the base state will decay back to the base state (i.e. the base state is stable).

We define $\epsilon = \frac{R - R_c}{R_c}$. For small positive perturbations ϵ , various types of spatial and temporal instabilities will set. This occurs when $Re(\sigma) = 0$, that is, where the growth rate of the perturbation is zero. The base state loses stability at R_c and the solution spontaneously loses symmetry which develops into a spatial pattern with characteristic wavelength $\lambda = 2\pi/k_c$.

We can categorize the spatial instabilities based on the most unstable wave vector, k_0 or a band of unstable wave vectors. $Im(\sigma(k_0)) = \omega_0$ gives information of temporal instability. There are three classes of behaviors based on the linear instability of the base state. Three main classes of pattern-forming systems [19]:

- Type I_s: stationary periodic, where $\omega_0 = 0$ and $k_0 \neq 0$
- Type I_o: oscillatory periodic, where $\omega_0 \neq 0$ and $k_0 \neq 0$
- Type III_s: oscillatory uniform, where $\omega_0 \neq 0$ and $k_0 = 0$

To further understand the instabilities, we can compute interactions between spatial eigenfunctions associated to the same eigenvalue [119]. Different classes will give rise to differences in amplitude equations, which govern patterns such as rise to ripples, square cells, and hexag-

onal cells. The dynamics of a solution are described by amplitude equations near equilibrium, but far from equilibrium, phase equations give a clearer picture.

For nonlinear systems, perturbation methods are provide good characterizations of spatio-temporal patterns near the base state because nonlinear effects are weak. However, when a system is driven far from equilibrium the control parameter R is far beyond the critical threshold and perturbation theory no longer gives an accurate description [109]. Nonlinear effects can play a larger role, the system is often disordered and non-ergodic. There is an absence of an established theoretical framework here, in part due to these challenges [59]. In this case, experiments and numerical simulations play a larger role in understanding bifurcations and the types of patterns that can occur.

3.3 Cubical complexes and PDEs

Numerical simulations a PDEs necessarily are discretized on a grid. To characterize the geometric structure of complex patterns that are formed, we turn to topological methods. This formulation lends itself naturally to building cubical complexes of sublevel sets parameterized by a threshold. The homology of this complex provides information about the geometric structure present. Computational homology has been used, for example, to study systems known to exhibit spatio-temporal chaos [43], and was proposed as an alternative way to compute Lyapunov exponents for chaotic systems. Homology has been used to compare experimental data with a common approximation method in the model for Rayleigh-Bénard convection [68]. Gameiro et. al track the evolution of Betti numbers as the system evolves in time to characterize complex microstructures that arise due to a process known as spinoidal decomposition [44]. These techniques rely on the first step of choosing an appropriate threshold. In some cases, like data arising from Rayleigh-Bénard convection, decomposing the domain into regions that represent upward flow and downward flow provides a natural division [68]. However, as [64] points out, homology groups can be very sensitive to small perturbations in the threshold. Considering

all threshold values—computing persistent homology— avoids this challenge and can provide a more complete summary of the geometric structure at various scales.

Persistent homology has been used to characterize Kolmogorov flow and Rayleigh-Bernard convection [66], to study force networks in compressed granular media [64] and to classify complex microstructures that arise in the Cahn-Hilliard equations [28]. Based on these initial findings, computational topology is proving to be an important tool in the quantification of complex structures.

However, before jumping in, we note a word of caution mentioned by Kaczynski et. al in [60]. In representing a continuous solution as a numerical approximation on the grid, we introduce numerical error and topological errors. To expand on this idea, we will look an example that is given in [60] and illustrated in Figure 3.2. We will represent two lines drawn to form a ‘V’ shape with a discretization of the domain. Take a discrete grid, for example 10×10 , if the line passes through a cell in the grid, give that pixel a value of one, otherwise, the pixel has a value of 0. Each pixel corresponds to a (potential) vertex in an associated cubical complex. Building the cubical complex on the pixels with value 0 (white pixels in the figure), we end up with a white pixel in the crook of the V is disconnected from the larger white region. Recall, as described in 1.5.1, when building the cubical complex, we choose to consider 4-neighbor connectivity for the sublevel set and 8-neighbor connectivity for the superlevel set. If we compute homology of the complex formed by the pixels with value 1, this will correspond to a superlevel set of the thresholded data, so we use 8-neighbor connectivity. The ‘V’ shape is homotopic to a point, so the homology group should be 0. However, we can see a spurious signal in the H_1 homology group.

We also introduce small numerical errors in the simulation of nonlinear PDEs. We should be aware of the possibility of these small topological errors as we proceed. However, in practice the influence of such features is small, especially if we are using persistent homology rather than just computing the rank of the homology group.

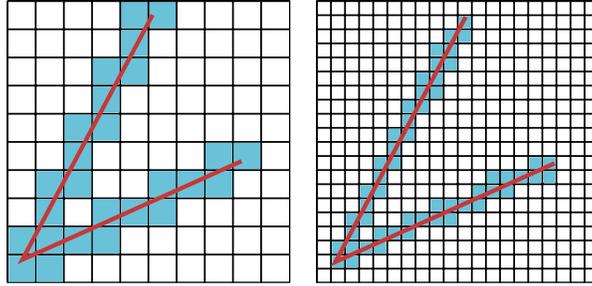


Figure 3.2: Consider a discrete approximation to the V-shaped red line. We will assign a pixel the value of 1 if the line crosses through the pixel (shaded above), otherwise the pixel value is 0. Even though the red V is homotopic to a point, and therefore should admit no H_1 , there is a small signal present due to the discretization. Recall that for sublevel sets, we consider 4-neighbor connectivity, so the white pixel in the crook of the V is disconnected from the larger white region. This is not a matter of resolution, if the resolution is increased, a small loop is still present. [60]

3.4 Anisotropic Kuramoto–Sivashinsky Equation

The Kuramoto–Sivashinsky (KS) equation is a partial differential equation that has been independently derived to model a number of pattern-forming systems driven far from equilibrium [90]. Its first appearance was to model flame front propagation [106]. It has found many applications in surface pattern-formation including surface patterning by ion-beam erosion [20, 83], epitaxial growth and instabilities related to electromagnetism [116], the formation of suncups in snowfields [112], and solidification from a melt [50]. The nonlinear term is often anisotropic in applications, giving the anisotropic Kuramoto–Sivashinsky (aKS) equation

$$\frac{\partial u}{\partial t} = -\nabla^2 u - \nabla^2 \nabla^2 u + \left(\frac{\partial u}{\partial x} \right)^2 + r \left(\frac{\partial u}{\partial y} \right)^2, \quad (3.1)$$

where $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$, and the real parameter r controls the degree of anisotropy.

The KS equation exhibits spatio-temporal chaos [101]. The one-dimensional KS equation undergoes a period-doubling cascade to chaos [107]. Smyrlis and Papageorgiou compute the continuum analogs of Feigenbaum’s universality constants α and δ for the one dimensional KS equation [107]. Feigenbaum predicted universal behavior for continuous flows of infinite dimension, and observed this phenomenon in Rayleigh-Bernard flows [66]. In our experiments, we focus on the two dimensional model.

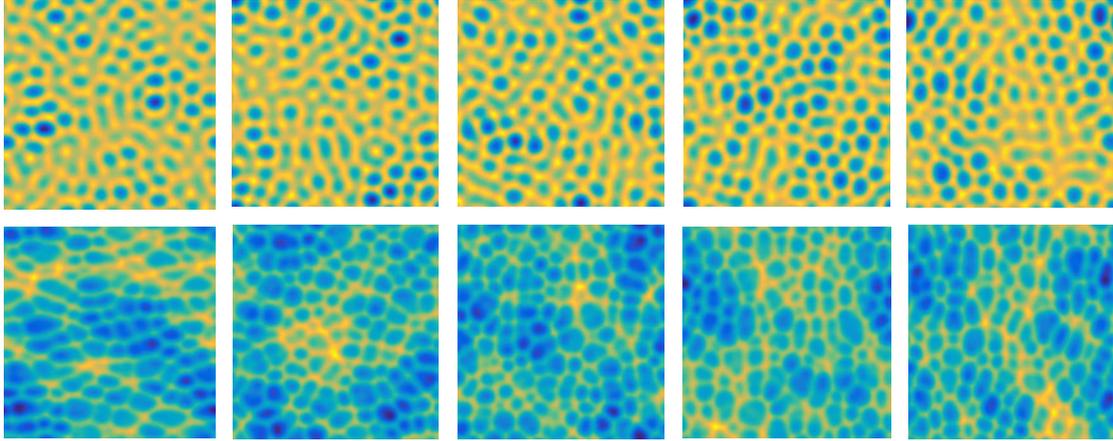


Figure 3.3: Plots of surface $u(x, y, \cdot)$ resulting from numerical simulations of the aKS equation (3.1). Each column represents a different parameter value: (from left) $r = 0.5, 0.75, 1, 1.25$ and 1.5 . Each row represents a different time: $t = 3$ (top) and $t = 5$ (bottom). By $t = 5$ any anisotropic elongation of the surface pattern has visibly stabilized.

The nonlinear terms stabilize the linear instabilities, establishing a dynamic steady state of irregular, bounded fluctuations [95]. The surface is composed of complex cells with a characteristic length that change in time, in a phenomena called chaotic bubbling [95], in which cells seem to split and merge in a random fashion.

The numerical technique for these simulations is as follows. We use a Fourier spectral method with periodic boundary conditions on a 512×512 spatial grid, with a fourth-order exponential time differencing Runge-Kutta method for the time stepping [16, 17]. Several examples of numerical simulations of the aKS equation for a range of parameter values (columns) and simulation times (rows) are shown in Fig. 3.3. The initial conditions for all of the simulations was low-amplitude white noise. We choose five parameter values, $r = 0.5, 0.75, 1, 1.25$ and 1.5 , generate thirty trials for each parameter. These parameters were chosen because visually the patterns produced are very similar (and therefore difficult to distinguish). The pattern produced by $r=1$ is isotropic.

3.4.1 Reduced Resolution and Variance Classification

In this project, our goal is to identify the anisotropy parameter for each simulation using snapshots of the surfaces $u(x, y, \cdot)$ as they evolve in time. Inference of the parameter using the

surface alone proves difficult for several reasons. The aKS equation exhibits sensitivity to initial conditions, so solutions that are initially close together diverge quickly. The spatial resolution of the numerical simulations provides a discretization of the surface on which we may consider applying machine learning techniques. However, even for our relatively small spatial grid (512×512) this means that the surface could be thought of as a vector in \mathbb{R}^{266144} . Because of the high dimensionality, we were unable to perform classification on the surface alone as a first attempt at classification. Dimensionality reduction was necessary. A coarse method of dimension reduction is generated by replacing blocks of grid elements in the spatial domain with their average value. The surfaces were resized in this way to a resolution of 10×10 , which gives a vector representation of the surface in \mathbb{R}^{100} . We applied a subspace discriminant ensemble to perform classification. It is not surprising that this method performed very poorly. (See Table 3.1).

The anisotropy parameter influences the mean and variance in the amplitude of the surface pattern. After each simulation, the surface is mean-centered to eliminate such a difference (which would not be accessible if using data as opposed to numerical simulations). It was thought that the variance in surface height could provide enough discriminating information. To test this idea, a normal-distribution based classifier was built on the variances of surface height. The normal distribution was fit to a set of training data consisting of $2/3$ of the simulations for each parameter. Classification was performed on the remaining $1/3$ based on the z -test for each of the different models. This is performed 100 times with random partitions of the data into training and testing sets. Accuracy is the average over all trials. Early in the formation of the pattern, the variance is not a discriminating feature (as seen in Fig. 3.4) and this method yields poor classification results. However, after the surface is allowed to evolve for a sufficient amount of time, the variance is a useful discriminator. We achieve a classification accuracy of 75% for later times. While this is a large improvement upon purely coarsening the spatial resolution, there is more structure in the pattern that will prove useful in classification.

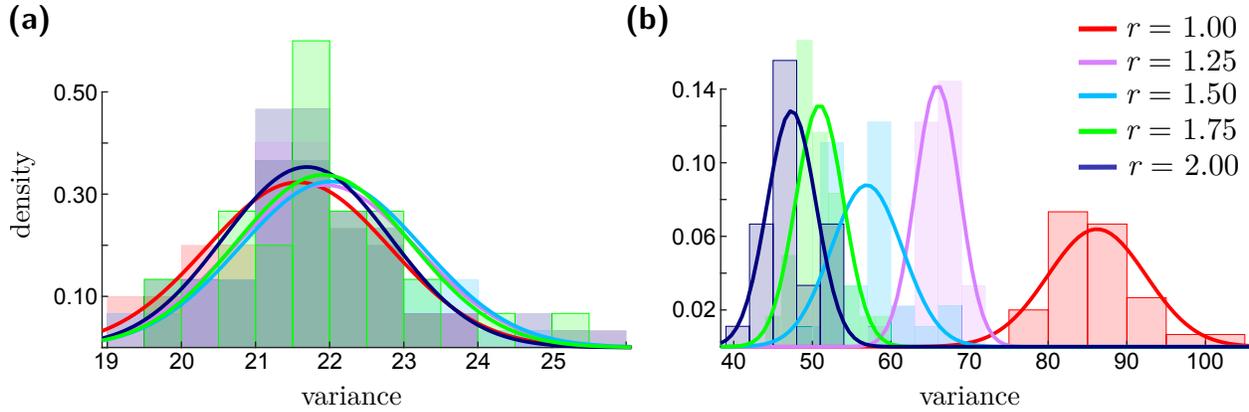


Figure 3.4: Histograms of the variances of surface heights for each parameter value, and the normal distribution fit to each histogram, for times (a) $t = 3$ and (b) $t = 5$. (Figure from [2])

Table 3.1: Classification accuracies for two simple classification methods on the solutions.

Classification Approach	Time t=3	Time t=5	Time t=10
Subspace Discriminant Ensemble, Resized Surfaces	26.0 %	19.3%	19.3 %
Variance Normal Distribution Classifier	20.74%	75.2%	77.62 %

3.4.2 Anisotropy Classification

The parameter r controls the anisotropy of the system. For $r > 1$, the pattern will be stretched in the y-direction and for $r < 1$, the pattern will be stretched in the x-direction. We compute a discrete approximation for the gradient of the solution and then compute the direction of the gradient between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. Figure 3.5 shows the resulting histogram of the gradient angles for a single example from each class. If we divide the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$ into 100 bins and count the number of gradient vectors falling in each angle range, we can build a feature vector for each example. Using this feature vector and a linear support vector machine³ we achieve a classification accuracy of 100 %. We maintain this accuracy even when we reduce the number of bins to 10.

³A linear Support Vector Machine (SVM) is a simple machine learning algorithm that finds the optimal hyperplane dividing vector data into classes. [67]

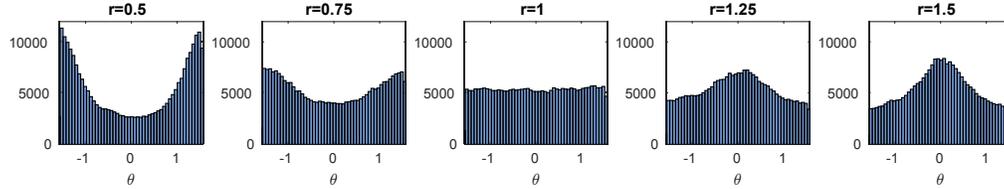


Figure 3.5: Histogram of the angle of gradient vector (mod π) for a single example of each parameter class at time step 15. The isotropy is clear for $r = 1$ as there is no preferential direction shown by the gradient vectors. Because of the differences in the distribution, it is reasonable to expect that the gradient vectors serve as a good classifier.

However, this technique relies on being able to distinguish the x and y directions, as there is a correspondence between the degree of anisotropy and the direction. When using simulated data, this is accessible, however, this may not be accessible if we are given real data. Homology is a topological invariant, and so it does not rely on the orientation of the domain. If one were to rotate the spatial domain by 90 degrees prior to computing the persistence, the resulting persistence diagram of the surface would remain unchanged. We will see that there is more information that is useful for classification contained in the topological structure of the solution.

3.4.3 Persistence of the aKS Equation

The sublevel set filtration reflects not only the variance of a surface, but also additional topological structure which may reveal other influences of the anisotropy parameter on the surface. We compute persistence of each surface using cubical homology and generate a persistence image for each snapshot in time of each surface. To generate the persistence images, we used a Gaussian distribution with variance $\sigma = 0.01$, a linear ramp function that was zero along the diagonal and one for the longest bar and beyond, and were able to reduce the resolution of the image to 20×20 . We tested classification persistence images generated with a range of resolutions and variances of the Gaussian. There is little variation for large sets of parameter choices. We include classification accuracies for persistence images formed when the resolution is held constant, and the Gaussian variance is changed in Table 3.3 and for persistence images formed with various resolutions at a constant variance in Table 3.4.

Table 3.2: Classification accuracies at different times of the aKS solution, using a subspace discriminant ensemble with a subspace dimension of half of the vector length and 200 learners. Also shown are the classification rates if we consider properly classified surfaces as ± 1 parameter class. Classification of times $t = 15$ and 20 result in accuracies similar to $t = 10$.

PIs	t=3 correct class	t=3 ± 1 class	t=5 correct class	t=5 ± 1 class	t=10 correct class	t=10 ± 1 class
H_0	60.7 %	100%	92.0%	100%	89.3 %	100%
H_1	68.0 %	97.3 %	84.7%	100%	78.0%	100%
H_0 and H_1	82.0%	100.0 %	87.3%	100%	82.7%	100%

To classify persistence images, we use a subspace discriminant ensemble [57]. The underlying assumption here is that different classes of data are generated by different Gaussian distributions by estimating a covariance matrix for the data. The parameters of the Gaussians are estimated for each class. To predict the class of new data, the misclassification cost is minimized. This process is performed repeatedly over random subspaces. The standard subspace dimension is half the dimension of the total space, so in this case, 50 when a single time step and single homological dimension are present. 200 iterations (or learners) is more than sufficient for the classification accuracy to level off. This model was trained with 5-fold cross validation. We achieve good classification results, especially early in the evolution of the surface. Concatenating H_0 and H_1 persistence images leverages information in both homological dimensions and leads to a gain in classification accuracy in this case. The high classification accuracy indicates that there are more subtle structural differences influenced by the anisotropy parameter.

Misclassification occurred nearly exclusively between adjacent classes, as shown in the “ ± 1 class” column of Table 3.2. This was the case for various times, resolutions and for classifiers using H_0 , H_1 and the concatenated data. This is consistent with the results of Dlotko and Wanner, who applied a similar process to simulations of Cahn-Hilliard equation [28]. In their experiment, they created averaged persistent landscapes for each parameter and each time step. Because the mean of persistence landscapes is unique [9] they used the mean as a representation of each parameter.

Table 3.3: Fixing a resolution of 20×20 and varying the variance of the Gaussian distributions, we see that classification accuracies are similar for a wide range of variances.

σ	Accuracy H_0	Accuracy H_1	Accuracy H_0 and H_1
0.0001	92.0 %	87.3 %	88.7 %
0.001	92.7 %	86.0 %	90.0 %
0.01	92.0 %	84.7 %	87.3 %
0.1	92.7 %	85.3 %	84.7 %

Table 3.4: Fixing the variance of the Gaussian at $\sigma = 0.01$ and varying the resolution, we see that a range of resolutions result in similar classification accuracies.

resolution	Accuracy H_0	Accuracy H_1	Accuracy H_0 and H_1
5	92.7 %	84.0 %	81.3 %
10	96.0 %	87.3 %	95.3 %
15	92.0 %	88.0 %	90.0 %
20	92.0 %	84.7 %	87.3 %
25	86.0 %	82.7 %	93.3 %

3.5 Application to Patterns Produced by Ion Bombardment

Shenoy, Chan and Chason studied the coupling between the surface topography and composition that arises during ion bombardment of a binary compound [104]. Bradley and Shipman (BS) extended this theory to include the effect of mass redistribution and the leading order nonlinear terms [6, 7, 105]. The BS equations govern the behavior of $u(x, y, t)$ and $\phi(x, y, t)$, the deviations of the surface height and surface concentration from their unperturbed, steady-state values. Following Bradley-Shipman, we have

$$\frac{\partial u}{\partial t} = \phi - \nabla^2 u - \nabla^2 \nabla^2 u + \lambda (\nabla u)^2$$

and

$$\frac{\partial \phi}{\partial t} = -a\phi + b\nabla^2 u + c\nabla^2 \phi + v\phi^2 + \eta\phi^3$$

for normal-incidence bombardment. Explicit expressions that relate the dimensionless constants a , b , c , λ , v and η to the underlying physical parameters may be found in Shipman et

al. [105]. We study solutions to the BS equations for various parameter values with classification based on parameters in mind. We will also use persistence as a summary of the topological structure to investigate the influence of nonlinear parameters on pattern formation and defects.

For some data sets, for example, the solutions to the aKS equations, a sublevel set filtration is a natural choice for computing persistence. Here, since the solutions we are considering display near hexagonal arrays of nanodots, it is natural to consider the Vietoris-Rips filtration of the xy -coordinates of the peak location of the nanodots. As we have seen in Chapter 5, this will give a measure of how close to perfectly hexagonal the arrangement of nanodots is. However, we notice that for some parameter values, long wave lengths influence the pattern. There are regions where the peaks of the nanodots are higher and areas where they are lower. This phenomena, shown in Figure 3.6 is important as the influence of these long wave lengths on pattern formation and defect annihilation is not well understood. For this reason, we will also compute persistence using the Rips filtration on the 3D coordinates of maxima and minima. Lastly, we include persistence generated with a discrete Morse filtration on the surface height as well.

Soft modes, or Goldstone modes are low-frequency modes effecting the long-range order of a pattern [87]. A mode with a “soft” dispersion relation so that $\omega(k) \rightarrow 0$ as $k \rightarrow 0$ is a Goldstone mode, or soft mode [70]. Spontaneous symmetry breaking with continuous order parameter degeneracy gives rise to these soft modes [70, 103]. The presence of soft modes reflect the absence of restoring forces for uniform displacements [19]. Their connection with defect formation is not well understood, it is believed that soft modes play a role in particularly in the resolution or lack of defects in ion bombarded systems [81]. Studying various parameter choices is a step in better understanding these effects.

In this investigation, we generate numerical approximations of the solutions with a Fourier spectral method with periodic boundary conditions and a fourth-order exponential time differencing Runge-Kutta method for the time stepping as the numerical technique [16, 17]. For all simulations of Eqs. (5.4) and (5.4), the initial conditions are low-amplitude white noise.

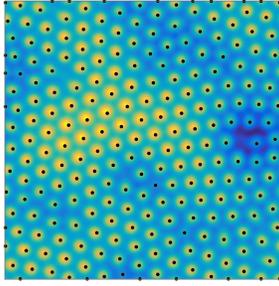


Figure 3.6: Plotted is the height function $u(x, y, \cdot)$ which is a solution to the BS equation,. The peak locations are marked with black dots. Here we can see the influence of long unstable wavelengths in the variation of the heights of these peaks.

Next, we generate persistence images from the resulting PDs. It should be noted here that when persistence images are generated, the longest bar in each homological dimension is chosen to be a parameter in the weighting function. This gives the longest bar a weight of one and scaling for shorter bars depending on the choice of weighting function. Further, the dimensions of the PI depend on the PDs over the entire data set. This approach is reasonable when the goal is classification of many examples in a set of data. When generating a PI from the PD computed from real data to compare to simulations, care should be taken in deciding appropriate parameters for the persistence image that are consistent with the classification model.

3.5.1 Varying Several Parameters

There are a number of parameters in the BS model that influence pattern type, characteristics and defects. We choose two values for λ and ν and three choices of b for the first classification task to demonstrate how topological characteristics may be used to distinguish parameter sets. The following two experiments will vary a linear and a nonlinear parameter (respectively) with a smaller increment between parameter choices.

First, we choose two different values of λ and ν which govern the formation of nanoholes or nanodots. There are 30 examples for each set of parameters, on a 256×256 spatial grid with approximately 40 time steps. We chose the parameters $a=0.25$ and $c=1$. $bc = \frac{(a+c)^2}{c} = 1.25^2 = 1.5625$ defines a critical bifurcation. We let $b = 0.9b_c, 0.95b_c$ and $0.99b_c$. Table 3.5 lists the parameters

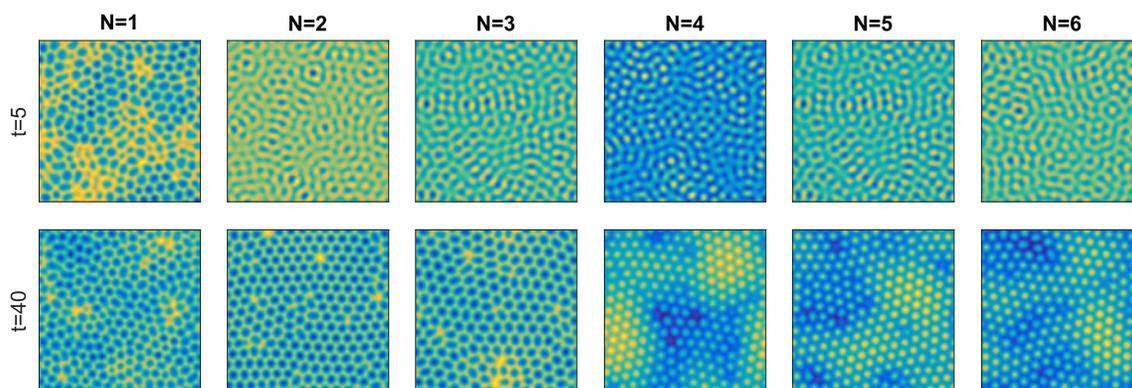


Figure 3.7: Examples of surfaces for the first parameter set at an early time ($t=5$) and a later time ($t=40$), after the hexagonal pattern has clearly emerged. One can see the emergence of nanoholes in the first three examples and nanodots in the second three examples. Both exhibit a nearly hexagonal lattice, rife with defects.

Table 3.5: Parameters for the first data set, we use N to distinguish classes.

N	b	λ	ν
1	0.9 bc	-0.5	2
2	0.95 bc	-0.5	2
3	0.99 bc	-0.5	2
4	0.9 bc	0	1
5	0.95 bc	0	1
6	0.99 bc	0	1

for each class. $\lambda < 0$ produces nanoholes rather than nanodots since $\nu = 2$ is not enough to overcome the negativeness of λ . This is seen in Figure 3.7.

We compute PH in three ways (with a 2D VR filtration on maxima locations, a 3D VR filtration on maxima and minima and a discrete Morse filtration on surface height). From the resulting persistence diagrams, persistence images are created, which will be used as feature vector for classification. All persistence images are generated with a bump function as the weighting function, starting with a minimum bar length of 0, up to a maximum of the longest bar over the entire data set for each homological dimension. The variance is fixed at $\sigma = 0.001$ and the resolution is chosen to be 20×20 .

Table 3.6: Classification accuracies for ion data set 1. Accuracies are reported for both the subspace discriminant ensemble and the bagged trees method for all three choices of filtrations. The final three columns give the percentage of examples classified in the correct class, or in one adjacent class.

	Subspace Discriminant Ensemble					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 H_1$	H_0	H_1	$H_0 \& H_1$
t=5, Morse	85.2 %	84.6 %	83.3 %	85.2 %	84.6 %	83.3 %
t=40, Morse	91.4 %	96.3 %	97.5 %	97.5 %	98.1 %	99.3 %
t=40, 3D VR	95.7 %	97.5 %	98.8 %	100 %	100 %	100 %
t=40, 2D VR	95.1 %	98.1 %	98.8 %	100 %	100 %	100 %
	Bagged Trees					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 H_1$	H_0	H_1	$H_0 \& H_1$
t=5, Morse	84.0 %	90.7 %	88.9 %	84.0 %	90.7 %	88.9 %
t=40, Morse	91.4 %	99.4 %	100 %	100 %	100 %	100 %
t=40, 3D VR	92.0 %	96.9 %	97.5 %	100 %	100 %	100 %
t=40, 2D VR	93.2 %	96.3 %	96.3 %	100 %	100 %	100 %

We will start by classifying at a single time step after the pattern has been allowed to emerge and develop. We use two classification algorithms, the subspace discriminant ensemble (as was used in Section 3.4) and the bagged trees ensemble.⁴ For the subspace discriminant ensemble, we use a subspace dimension of 100 for H_0 and H_1 PIs and 200 for the concatenated vectors. The number of learners is the number of times the classification on a random subspace is repeated. We are not concerned with optimizing the number of learners, so we choose 100 learners for each ensemble since this seems to be well past the threshold where each additional learner has a large effect on the overall classification. Even with this many learners, classification is complete in less than a minute. We employ 5-fold cross validation in our model.

The results of the classification are summarized in Table 3.6. In this case, PIs generated from all filtrations performed very well. Using concatenated H_0 and H_1 PIs increases the classification accuracy. There are small differences between the two different machine learning algorithms and between the different filtrations, however, the difference is small. Interestingly,

⁴The method of bagged trees, is a **Bootstrap Aggregation** method, a subspace of interest is chosen at random. On each subspace, a decision tree classifier is built and “votes” on the appropriate class for each example. Final classification is decided by totaling the “votes” for each example. [67]

the 2D VR filtration performs slightly better than the 3D VR filtration, indicating that in this case there is not additional information useful for differentiating classes that is contained in the location of the valleys, nor in the 3D location of the peaks. In general, this does not rule out the possibility that useful discriminating information is contained in the heights of the maxima, rather than solely in their planar location. In the next two experiments, we retain the three types of filtrations.

We also report the percentage of examples classified in the correct class, or in one adjacent class. For example, a surface whose actual class is $N = 5$ but is classified as $N = 6$ is counted in this column. No examples of the first three classes were misclassified as a member of any of the second three classes or vice versa, indicating that there is a clear separation between the samples exhibiting nanodots and nanoholes. This is unsurprising given the obvious differences between the two. This difference is reflected in the persistence diagrams as well. (See Figures 3.8-3.10.) The H_1 points of the VR filtrations are much more tightly clustered for $N = 4 - 6$, which indicates the regularity of the peak structures (nanodots). For classes 1-3, H_1 bars seem to have a larger variance in birth scale and a smaller variance in the death scale. This is capturing the fact that most nanoholes at these parameter values are approximately the same size.

To begin to understand the features that are aiding in classification, we first consider the average variance of the surface as well as the number of bars in the 2D VR PDs. The number of H_0 bars is equivalent to the number of nanodot in the pattern. This is reported in Table 3.7. While there are notable differences between the first three classes and last three classes, the number of bars and the variance are not enough to be solely responsible for high classification accuracy. In fact, class 5 and 6 have the same average variance and the average number of bars are close to each other. This indicates that PH is detecting more subtle geometric properties of the pattern.

Good classification accuracies in this experiment, coupled with the fact that misclassification occurred in adjacent classes indicates that the topological structure summarized in a PD is useful for distinguishing the underlying parameters which drive pattern formation. We proceed

Table 3.7: For the time step $t=40$, we compute the average number of H_0 and H_1 bars in the 2D Vietoris-Rips PD over all examples as well as the average surface variance. Similar variance for $N = 1 - 3$ and $N = 4 - 6$ indicates that the classifier is picking up on more properties than solely a difference in variance. Similarly, differences in the numbers of H_1 bars alone (which for the 2D VR filtration corresponds to the number of nanodots) is not enough to account for good classification. This indicates PH is picking up on more subtle geometric differences in the pattern.

N	# H_0 bars	# H_1 bars	variance
1	417.2	200.8	2.24×10^{-5}
2	440.1	196.7	8.87×10^{-6}
3	369.2	164.3	3.52×10^{-6}
4	261.1	126.6	3.00×10^{-3}
5	252.0	124.3	6.18×10^{-4}
6	246.4	120.3	6.18×10^{-4}

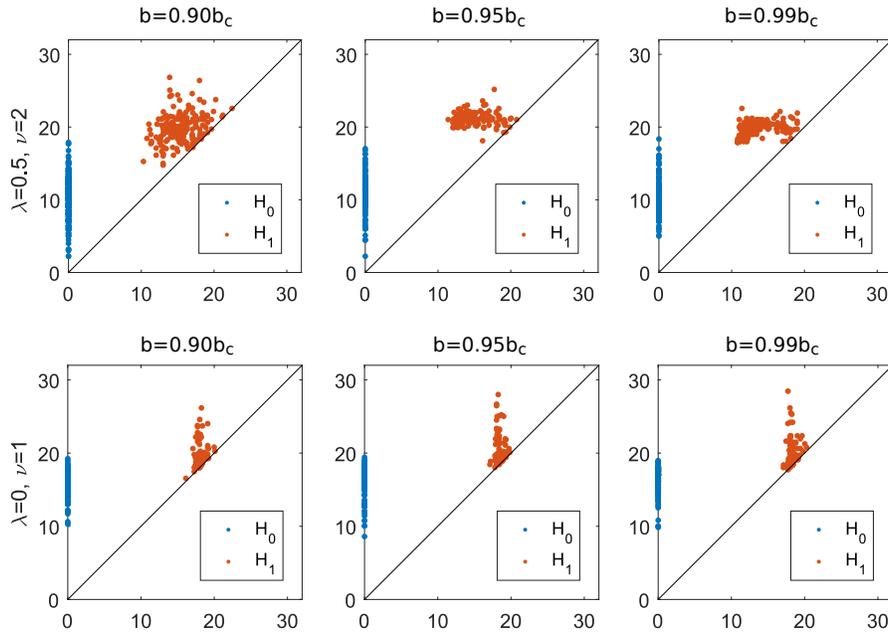


Figure 3.8: Example of PDs for set 1 generated from a Rips filtration on the 2D point cloud of maxima locations only, taken at the last time step in the simulation, $t=40$

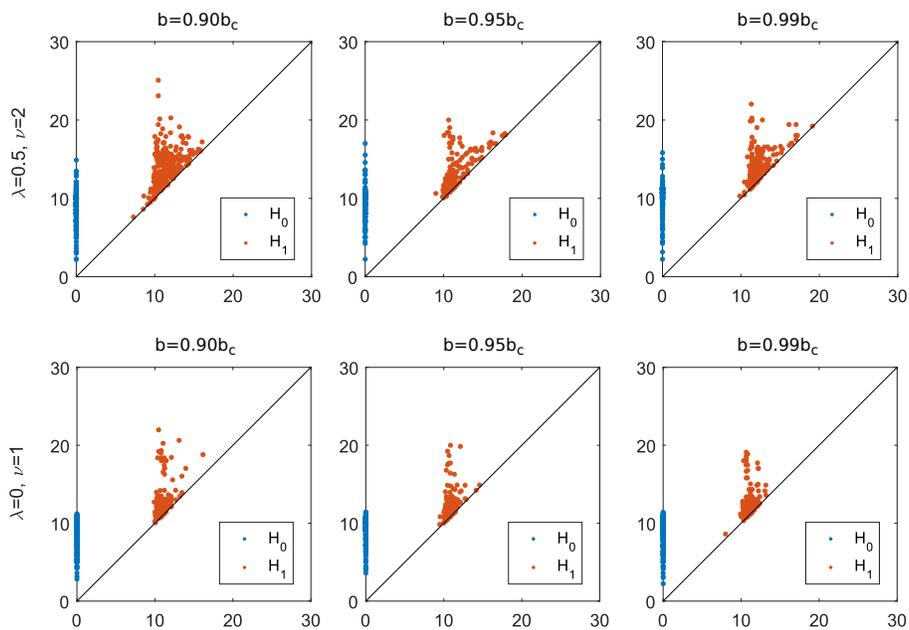


Figure 3.9: Example of PDs for set 1, generated from a Rips filtration on the 3D point cloud of maxima and minima, taken at the last time step in the simulation, $t=40$

by varying a single parameter while holding all others fixed. In Section 3.5.2 we vary a linear parameter and in Section 3.5.3 we vary a nonlinear parameter. We will classify at several times along the evolution of the pattern. In the last section, we consider one method of including temporal evolution information in the classification.

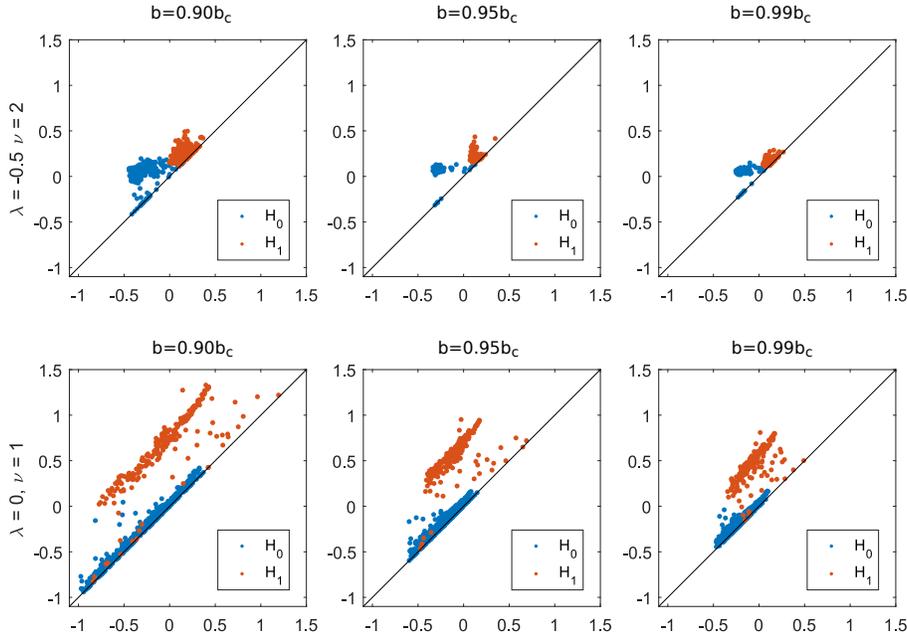


Figure 3.10: Example of PDs for set 1, using a Morse sublevel set filtration, taken at the last time step in the simulation, $t=40$. One can clearly see the difference between the examples shown on the top row, where λ is negative and the surface develops nanoholes, indicated by the loner barlengths for H_0 . The nanodots are most clearly indicated by the characteristic barlength present in the H_1 diagram.

3.5.2 Varying a Linear Parameter

First, we vary the linear parameter b with a small incrementation and will try to classify the resulting PIs by class. This amounts to identifying the parameter b . Fix $\lambda = 0$ and $\nu = 1$, which is in the parameter regime where we expect near hexagonal nanodots. b is varied from $0.9b_c$ to $0.99b_c$ incremented by 0.1. The linearly selected wave length does not depend on b , and therefore, we expect that the average inter dot distance to be the same across all examples. (This means that a difference in the number of H_1 bars is not the primary feature contributing to classification.) We expect more defects to be present for lower values of b . Visually, the difference between the chosen values of b are small and difficult to distinguish, which makes this an interesting machine learning task! Figure 3.11, show an example of each parameter value at several points in time in the pattern evolution.

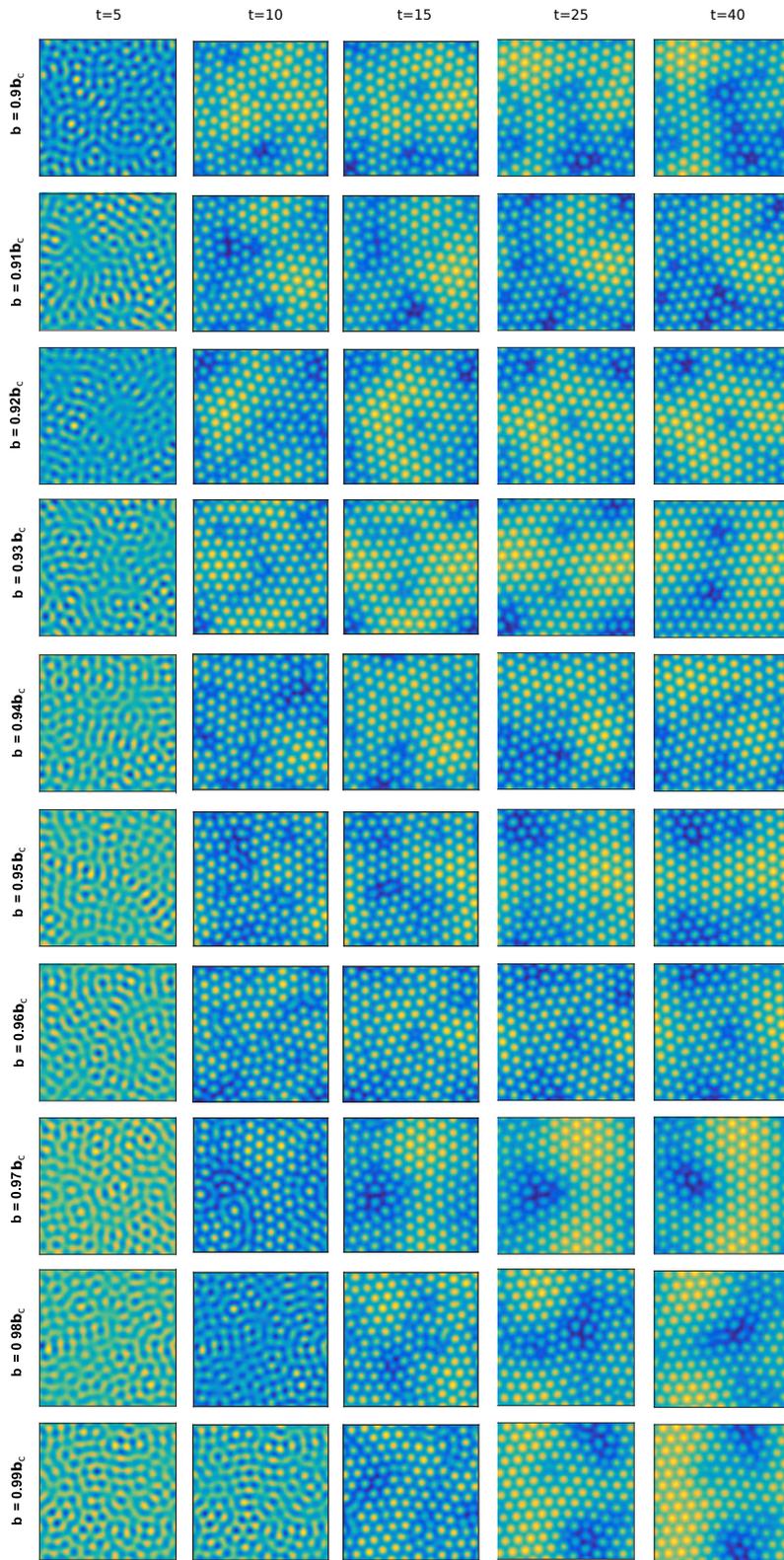


Figure 3.11: Examples of the solutions to the BS equations for values of b , approaching the critical value b_c . Shown are three moments in time; $t=5, 10, 15, 25$, and 40 . Note that for higher values of b , the pattern takes longer to set in.

We use the same PI parameters and the machine learning algorithms as in the previous experiment. All persistence images are generated with a bump function as the weighting function, starting with a minimum bar length of 0, up to a maximum of the longest bar over the entire data set for each homological dimension. The variance is fixed at $\sigma = 0.001$ and the resolution is chosen to be 20×20 . A subspace discriminant ensemble and bagged trees ensemble, with 100 learners and 5-fold cross validation are used to classify the PI vectors.

Preliminary classification results are shown in Table 3.8. The VR filtration on two and three dimensional points perform very poorly, though there is a preference towards the correct class. The percentage of points classified in the correct class ± 1 is much higher. However, classification is still not very good when compared to the classification accuracies achieved by the Morse filtration. We will continue our investigation with the Morse filtration.

Again, we normalize the surface to have a mean height of 0 and a variance of 1 and perform classification. The data is mean centered already (in all experiments), so this is a way to remove any discriminating information that variance may have. The results are reported in Table 3.9. Classification accuracies are two to three times higher than if examples were assigned a class randomly. Further, there is preference towards the correct class, though the model may miss by a class or two. This indicates that there is some topological structure that plays a role in classification, however the variance of the surface overall does play a role in aiding classification here.

Next we consider several different moments in the evolution of the pattern. Notice in Figure 3.11, at $t = 5$ the pattern is just starting to emerge. By $t = 25$ a nearly hexagonal array of nanodots has emerged but is rife with defects. By a later time $t = 40$, the pattern has resolved to varying degrees, however, there are still many defects. Classifying at each of these moments, using the Morse filtration only, we achieve reasonable accuracy, especially if adjacent classes are included. The results are reported in Table 3.10. We notice that for early time, H_0 is instrumental in achieving good classification, however at later times, its utility had dropped significantly. This is due to the fact that H_0 picks up the average inter dot spacing. Since this is governed by

Table 3.8: Summary of classification accuracies as b varies. We consider a single moment in time, $t=25$. In this instance, since most examples exhibit a high degree of hexagonal order, 2D and 3D VR filtrations do not serve as a good classifiers.

	Subspace Discriminant Ensemble					
	Correct Class			Correct Class \pm		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
2D VR	20.0 %	10.3 %	14.3 %	59.7 %	44.7 %	58.7 %
3D VR	16.3 %	9.7%	12.0 %	58.0%	42.0 %	63.3 %
Morse	33.7 %	86.7 %	73.0%	84.0 %	94.7 %	85.0 %
	Bagged Trees					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
2D VR	16.3 %	16.0 %	14.3 %	62.7 %	54.7 %	55.7 %
3D VR	14.3 %	13.0 %	16.3 %	63.3 %	49.0 %	57.7 %
Morse	22.0 %	67.3 %	66.7 %	34.7 %	99.0 %	99.0 %

Table 3.9: Classification is performed on surfaces that are normalized to have a mean of 0 and a variance of 1. This is done for time $t=25$. Classification accuracy drops significantly on surfaces that are normalized, however is still better than randomly assigning classes.

	subspace discriminant ensemble			bagged trees		
	H_0	H_1	$H_0 H_1$	H_0	H_1	$H_0 H_1$
3D VR	14.0%	13.0 %	18.3%	10.0%	15.0%	13.3%
Morse	24.7 %	17.3 %	20.3%	20.0%	20.7 %	23.3%

Table 3.10: We perform classification for an earlier time and a later time using PIs generated from the Morse filtration.

	Subspace Discriminant Ensemble					
	Correct Class			Correct Class \pm		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
t=5	63.0%	69.0%	71.0%	88.3%	90.0%	89.3%
t=25	33.7 %	86.7 %	73.0%	84.0%	94.7%	85.0%
t=40	31.3 %	86.0%	76.7%	76.7 %	96.0%	97.3%
	Bagged Trees					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
t=5	71.7%	74.7%	76.7%	96.3 %	93.0%	94.3%
t=25	22.0%	67.3 %	66.7%	34.6 %	99.0%	99.0%
t=40	26.0 %	71.7 %	67.3 %	68.7 %	99.0 %	98.0%

the linearly selected wavelength, which does not depend on b , we would not expect there to be big differences between the classes in this regard. The H_0 signal that is aiding in classification is picking up on the defects in the surface. Examples of the associated persistence diagrams are shown in Figure 3.12.

Classification using PIs generated from a Morse function classify an example in the correct class well. If we consider all points that are assigned the correct value of $b \pm 0.01b_c$, we achieve nearly perfect accuracy. This is remarkable since for this range of b , the BS equations are not very sensitive to b . We turn our attention now to varying a nonlinear parameter.

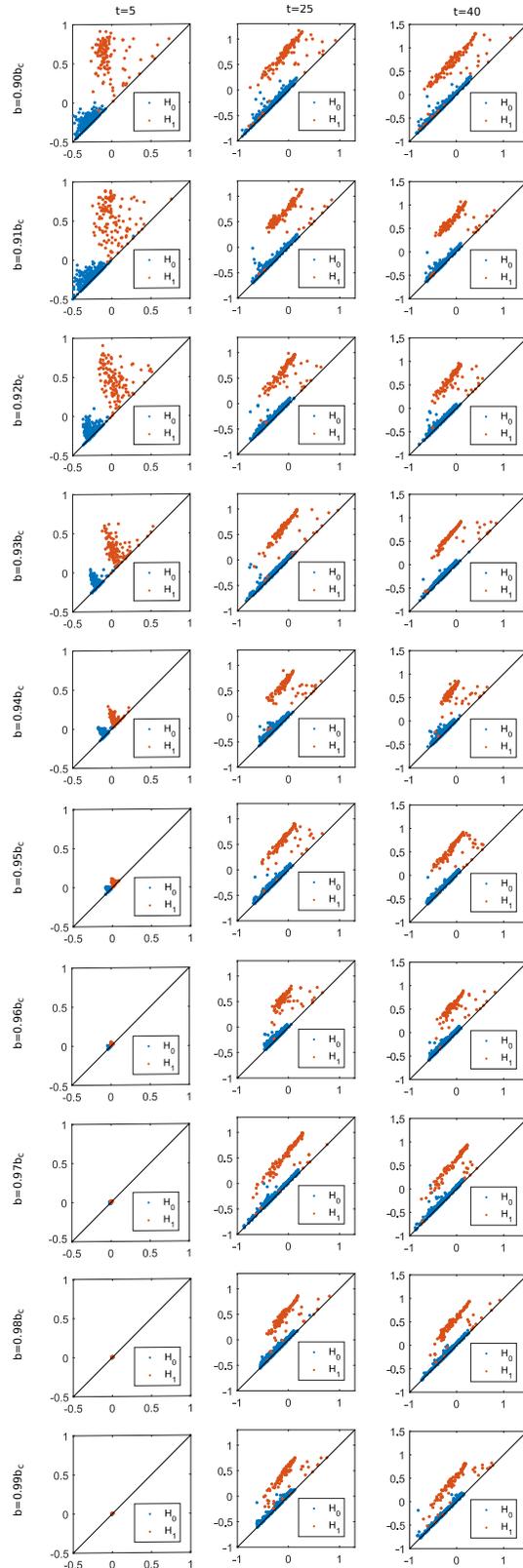


Figure 3.12: Examples of PDs generated using the Morse filtration for times $t=5$, 25 and 40 . It is easy to see that the spread of the persistence points for lower values of b contributes to a higher classification accuracy at early times. As the hexagonal pattern forms and emerges, differences become more subtle.

3.5.3 Varying a Nonlinear Parameter

We repeat similar classifications on data varying ν , letting $\nu = 0.5$ to 1.7 , incremented by 0.1 . Similar to our previous experiment, we choose $a = 0.25$, $c = 1$, $b = 0.95 * b_c$ and $\lambda = 0$ which produces near hexagonal arrays of nanodots, with defects. Varying ν will have a greater influence on defects through effecting the soft mode. We generate 30 examples for each choice of ν , each one of which has 200 time steps. This provides time to observe initial pattern formation and for the annihilation or continuation of defects. We observe that for some examples defects, like ripples, remain through the entire duration of the simulation. For other examples in the same parameter class, the pattern resolves to a nearly perfect hexagonal lattice. The variation of the pattern in a single class will make this task more challenging.

Again we use two methods of classification, a subspace discriminant ensemble and the bagged trees ensemble on PIs generated with a bump function as the weighting function, $\sigma = 0.0001$ and a resolution of 20×20 . The classification accuracies are reported in Table 3.11. In this case, we notice significantly better classification when using the Morse filtration. Misclassifications occurred primarily as an example being classified in an adjacent class, that is the ν value assigned by the model was ± 0.1 .

We investigate simple geometric properties such as the variance of the surface to decide if PIs are capturing more subtle topological information. The average variance across all examples for each parameter at time 25 is shown in Table 3.12. Variance increases as ν increases and could be playing a significant role in the classification.

To decide how much influence variance is having on the classification, we normalize each solution to have a mean of 0 and variance 1 before computing PDs. We classify the resulting PIs. This is reported in Table 3.13. Remarkably, the classification accuracy for the PIs generated with the Morse filtration increased. This indicates that there is more subtle topological information contained in the pattern that is captured by persistence and enables good classification.

When classifying on the normalized surface using the 3D VR filtration, classification accuracy is much better for $\nu \geq 1.2$, classifying nearly all examples in the correct class ± 2 . For

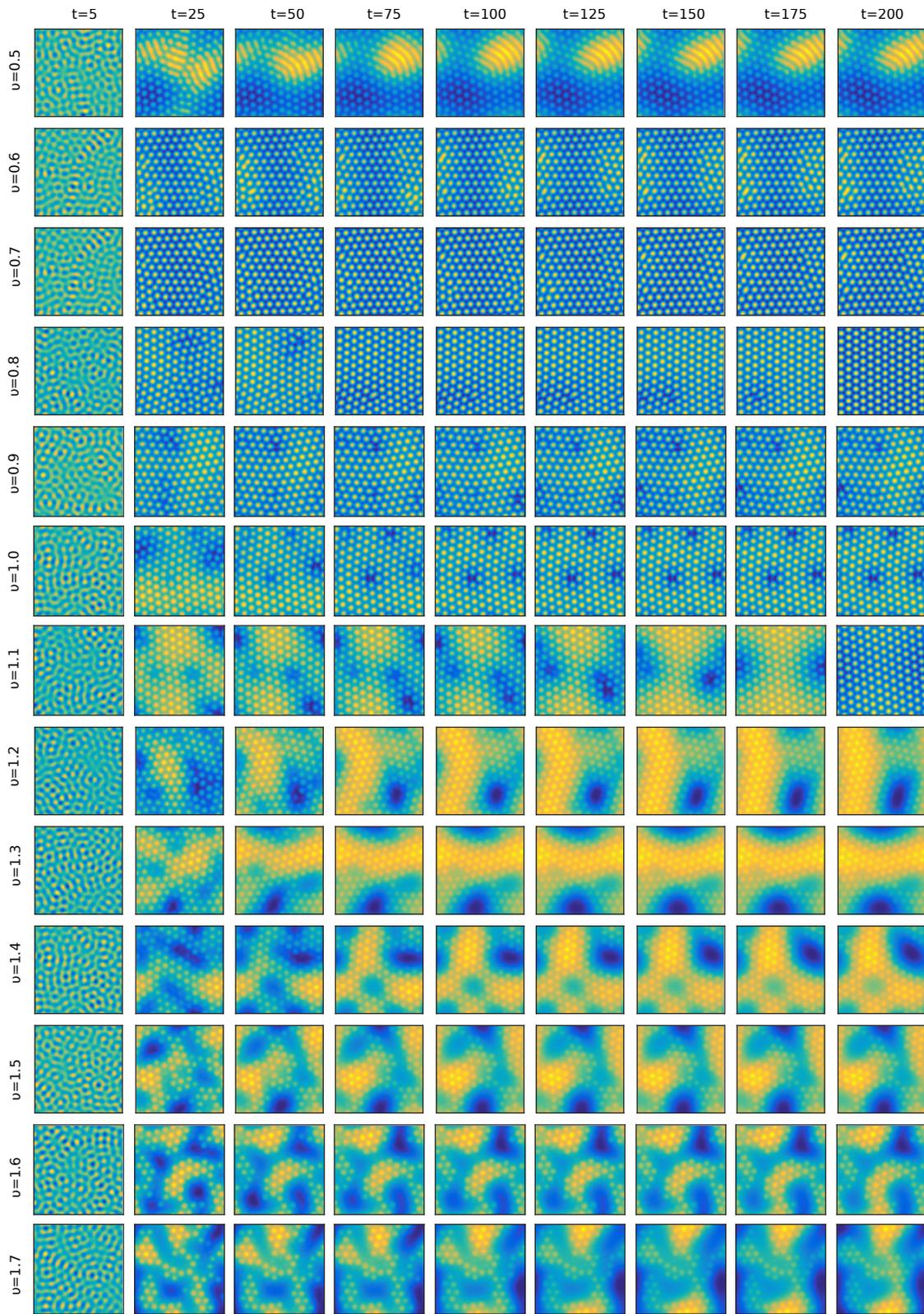


Figure 3.13: Examples at moments in time as v is varied. After a long time, for some parameter values, the pattern has resolved and for others, long wave defects have become pervasive.

Table 3.11: Summary of classification accuracies as the nonlinear parameter ν is varied. In this experiments, Morse PIs still perform better, however the difference is not as clearly pronounced.

	Subspace Discriminant Ensemble					
	Correct Class			Correct Class \pm		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
t=25, Morse	52.7 %	74.6 %	71.3 %	87.1 %	99.2 %	96.1 %
t=25, 3D VR	40.0 %	30.8 %	32.8 %	67.4 %	76.9 %	74.6 %
t=25, 2D VR	38.2 %	33.6 %	37.9 %	87.9 %	72.0 %	78.7 %
	Bagged Trees					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
t=25, Morse	64.9 %	70.8 %	74.4 %	91.8 %	98.2 %	97.9 %
t=25, 3D VR	37.7 %	34.6 %	35.1 %	69.0 %	69.0 %	74.6 %
t=25, 2D VR	35.6 %	37.7 %	36.4 %	61.1 %	70.0 %	70.0 %

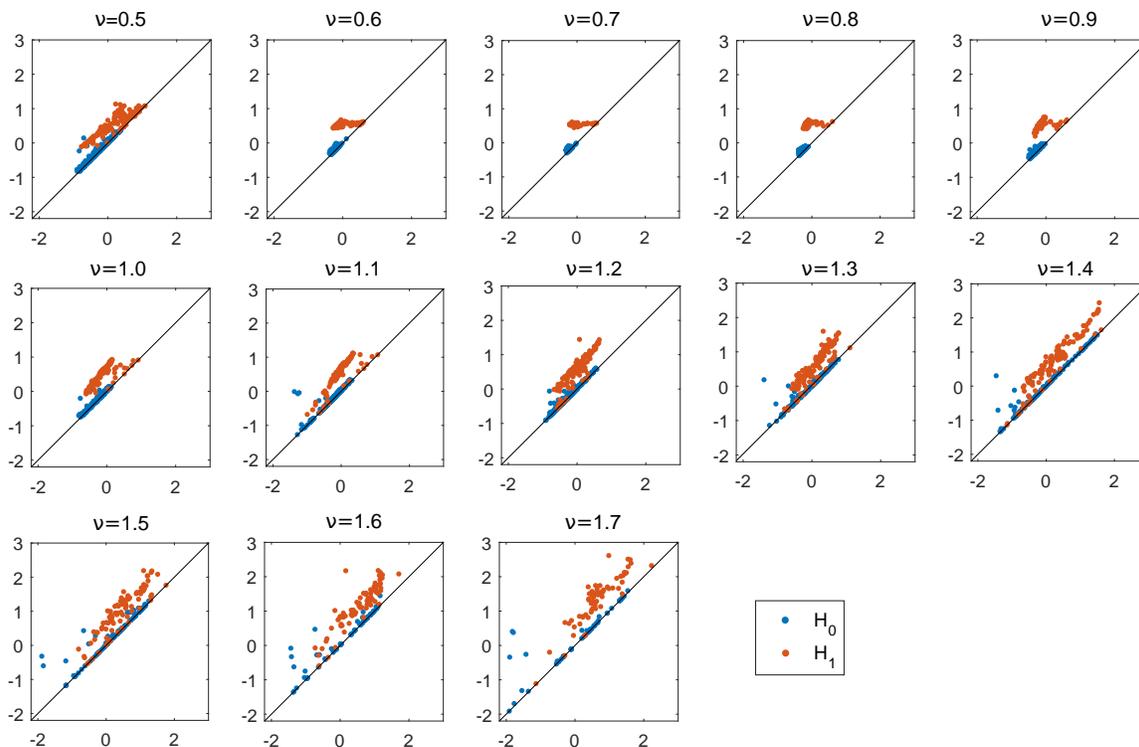


Figure 3.14: Examples of PDs for $t=25$ computed using the Morse filtration as ν is varied. Notice the tight clustering of persistence points for $\nu \in [0.6, 0.9]$. Longer H_0 and H_1 bars emerge for higher values of ν due to the large defects.

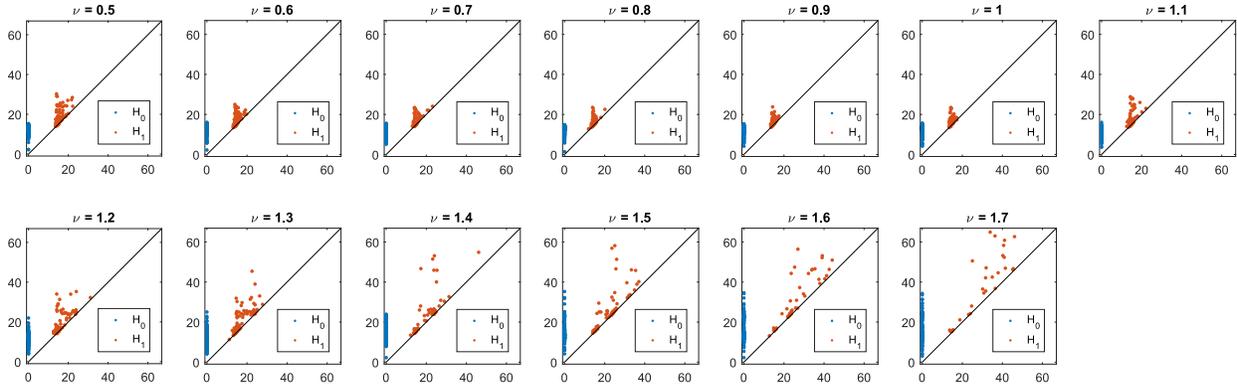


Figure 3.15: Examples of PDs computed using the 3D VR filtration on point cloud of maxima and minima as ν is varied, $t=25$. The effects of the defects are particularly striking for higher values of ν , indicated by long H_0 and H_1 bars. Classification is more accurate for $\nu \geq 1.2$, however for smaller values of ν , this filtration loses some discriminating information. In fact, classification is poor for $\nu < 1.2$.

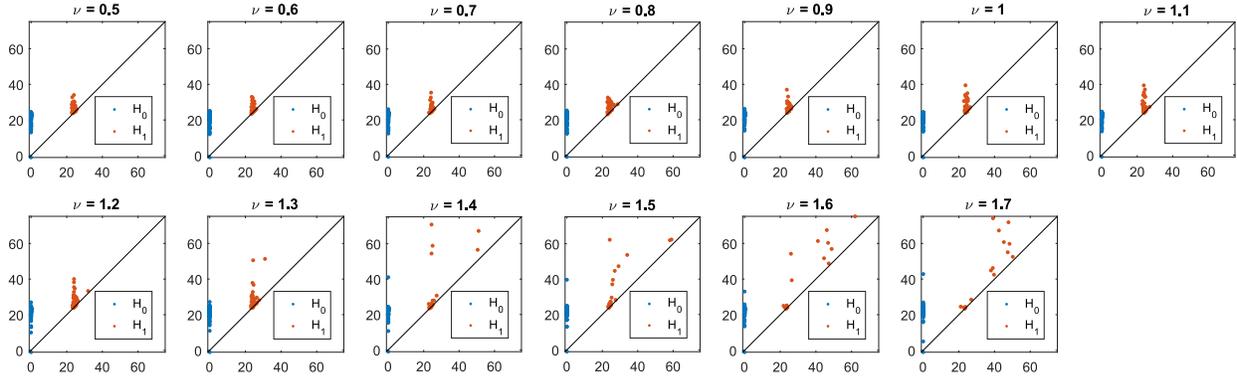


Figure 3.16: Examples of persistence diagrams computed using the 2D Rips filtration on xy-maxima locations, $t=25$.

Table 3.12: Computed are the average variances across all trial at $t=25$. As ν is increased, the average variance increases.

ν	avg # peaks	avg variance
0.5	148.3	9.0×10^{-4}
0.6	148.0	1.04×10^{-4}
0.7	146.9	7.95×10^{-5}
0.8	146.4	8.64×10^{-5}
0.9	146.2	1.40×10^{-4}
1.0	146.6	4.08×10^{-4}
1.1	147.1	1.7×10^{-3}
1.2	146.3	8.34×10^{-3}
1.3	137.8	3.08×10^{-2}
1.4	122.3	6.1×10^{-2}
1.5	109.2	6.95×10^{-2}
1.6	97.2	6.94×10^{-2}
1.7	85.0	8.58×10^{-2}

$\nu < 1.2$, classification accuracy is poor and there is less preference towards the correct class. In this regime, larger low regions without any nanodots have emerged, which would show up as longer H_1 bars. These defects create a higher variance of the surface overall, so when the surface is normalized, the height of an average nanodot is scaled differently that for lower values of ν for example.

Table 3.13: Classification accuracies of normalized data.

	Subspace Discriminant Ensemble					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
Morse	67.9 %	76.7 %	74.9%	93.3 %	98.2 %	99.2 %
3D VR	54.9 %	53.8 %	55.1%	80.5 %	89.0 %	90.0 %
	Bagged Trees					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
Morse	69.7%	68.7%	72.8%	95.1 %	96.4 %	98.9 %
3D VR	56.2%	53.6%	57.9%	83.7 %	75.1%	88.7%

So far we have considered only a single time step, but these patterns evolve dynamically. We will investigate whether incorporating several time steps into a classifier can increase clas-

sification. First, we discuss scaling options. The height of an individual pixel is the longest barlength (for each homological dimension over the entire data set of interest), divided by the chosen resolution of the image. The width of a pixel is the largest birth time (for each homological dimension, over the entire data set), divided by the resolution of the image. This means that often a pixel represents a rectangular region of the original persistence diagram, and may have a different size for different homological dimensions. This is of no real consequence if we use PIs as feature vectors for machine learning tasks, even if we concatenate vectors representing different homological dimensions. However, if we wish to sum or average PIs, we must ensure that each pixel has the same spatial extent in the metric of our original data set.

Along this line of reasoning, we incorporated several time steps in our model with two different methods of scaling. The first, listed as “indep. class scaling” in Table 3.14, was created by generating the PIs for each time individually. This is equivalent to choosing two sets of PIs (with the same filtration) for the previous experiments and concatenating for each example, and is akin to normalizing. In this case, we still take the bump function parameter to be the length of the longest bar over the entire data set to be weight 1. In general we have noticed that the maximum barlength increases as t increases, likely due to larger defects becoming locked in. The other way that one could scale is listed as “entire set scaling”, which means that the pixel size is chosen by taking the longest bar and latest birth time (for each homological dimension) over the entire set of interest, which includes multiple times. We highlight one example in particular where this affected classification. For times 10,15 and 20, classification was performed on PIs generated with both types of scaling. Scaling each class individually performed better than scaling over the entire data set, in particular for H_1 , which is consistent with the result that in this case, PIs produced from normalized data classified better.

Including an additional time almost always improved classification accuracy, with the largest increase occurring when the times are close together. This would seem to indicate that v influences the way these lower regions cause by the soft mode either deepen and persist or even-

tually are invaded with nanodots. The dynamics of this process are more clearly picked up on over shorter time differences.

In the future, we hope to investigate incorporating the dynamic component of the pattern into our classification. Cubical complexes are defined in higher dimensions, [117], for example, defines a fast algorithm for computing higher dimensional cubical complexes. This is a very natural way to incorporate the temporal component into the filtration. Our initial findings in this section indicate that the dynamic evolution of the surface to being able to identify driving parameters. The next step in this project is to compute PDs on 3D data; two spatial dimensions and a temporal dimension and use the PIs generated in this way to classify on parameters and further investigate the effects of parameters on the soft mode defects and defect annihilation.

Table 3.14: Classification results for several different combinations of times, varying ν . PDs are computed with a Morse filtration. The scaling method in each instance is listed. “indep. class scaling” means that PIs were generated for each time step individually, and then the PIs were concatenated. “entire set scaling” indicates that PIs were generated for the entire set all at once. Classification of a single time, $t=25$ is listed again for comparison.

	Subspace Discriminant Ensemble					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
$t=25$	52.7 %	74.6 %	71.3 %	87.1 %	99.2 %	96.1 %
$t=25/50$ indep. class scaling	58.5%	75.9 %	78.2%	83.6 %	98.9 %	96.9 %
$t=10,15,20$ indep. class scaling	67.2%	84.4%	73.3%	95.3 %	98.4 %	91.2 % %
$t=10,15,20$ entire set scaling	48.5%	70.3%	75.4%	77.7 %	90.0 %	% 91.3
$t=10,11$ entire set scaling	66.7%	90.5%	92.1%	91.5 %	98.7 %	97.4 %
$t=15,16$ entire set scaling	62.8%	90.0%	93.1%	85.6 %	97.4%	97.7 %
$t=15,16,17,18$ entire set scaling	85.9%	98.2%	96.9%	95.8 %	100 %	99.2 % %
	Bagged Trees					
	Correct Class			Correct Class ± 1		
	H_0	H_1	$H_0 \& H_1$	H_0	H_1	$H_0 \& H_1$
$t=25$	64.9 %	70.8 %	74.4 %	91.8 %	98.2 %	97.9 %
$t=25/50$ indep. class scaling	65.1 %	82.3%	78.5%	90.7 %	100 %	99.4 % %
$t=10,15,20$ indep. class scaling	66.9 %	85.6%	86.4%	98.4 %	100%	100 % %
$t=10,15,20$ entire set scaling	42.6%	67.7%	66.9%	67.7 %	89.5 %	87.4 %
$t=10,11$ entire set scaling	66.3%	87.4%	85.4%	91.2 %	97.7 %	96.6 %
$t=15,16$ entire set scaling	70.8%	91.8%	87.2%	91.5 %	97.7 %	96.9 %
$t=15,16,17,18$ entire set scaling	86.2%	96.9%	96.7%	92.8 %	99.7 %	98.4 %

Chapter 4

Characterization of Surface Properties

4.1 Introduction

In this chapter we discuss several applications of PH to characterizing geometric properties of surfaces. The first characterizes surface roughness of melting snow fields in mountainous terrain. Persistence is able to capture the small scale features of the snow as well as the large scale features of the terrain. We discuss a persistence-based method of investigating the roughness of snow field from collected data. The second section develops a PH based statistic for comparing peaks and valleys in a laboratory experiment in which irradiated surfaces exhibit a pyramid-dominated pattern or an anti-pyramid dominated pattern, depending on system parameters.

4.2 Snow Roughness

In this section, we use persistence to gauge multiscale surface roughness, specifically the roughness of snowfields as they melt. The snowpack surface is spatio-temporally complex. The geometry of the surface can undergo dramatic changes at various length scales due to snow ac-

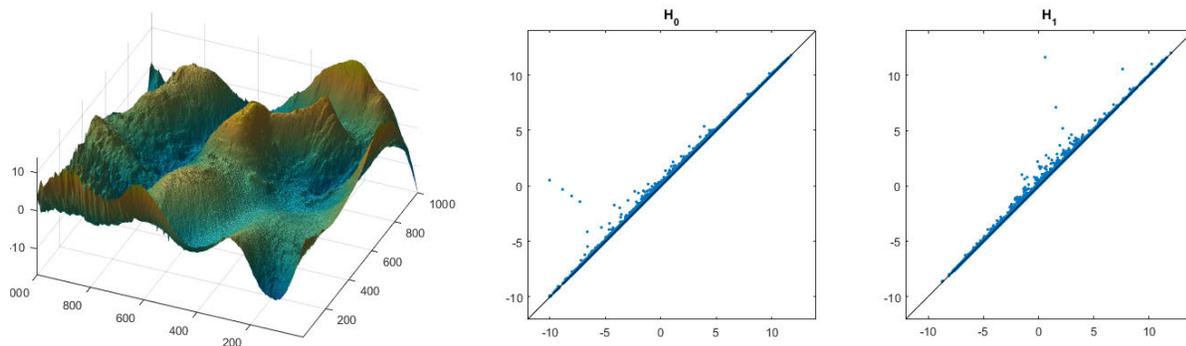


Figure 4.1: Plot of the entire snow surface and the associated PDs. Topographical features, like mountain peaks, appear as long bars in the PD. Smaller scale rough features appear as smaller bars.

cumulation, terrain features and wind. Deems et al. show that snow depth has a fractal distribution over a wide range of length scales [25]. Ablation hollows, or “suncups” form on the surface of snowfields in the summer and effect the albedo of the snow surface [112]. Albedo, which can be loosely described as the “whiteness” of the snow, is the ratio of radiation that is reflected from a surface to the incident radiation. Tiedje et al. model suncups with a Kuramoto-Sivashinsky like equation [112]. The result is a pattern that is a constantly evolving, chaotic pattern with a fixed amplitude. As the surface changes, albedo, wind resistance, energy exchange and melt-water production are all affected [55].

Snowpack is at the interface of between the Earth and the atmosphere and influences the movement of air [37]. Surface roughness influences energy exchange, heat transfer and melting [55]. Characteristics of the snowpack surface are important input variables in models so accurate estimates of these parameters are needed. There are a variety of roughness measures, autocorrelation, random roughness, fractal dimension, geometric roughness length, curvature, and power spectral density that are all used to characterize snow surface roughness. A comparison is provided in [37].

We introduce two different methods of creating a persistence-based roughness metric. It is common to use one-dimensional cross sections in analyzing structure formation in snowfields [80] because wind direction causes anisotropy. In this spirit, we start with one-dimensional sublevel set persistence. We treat each row or column of the data matrix as a 1D function, and use a sublevel set filtration to sweep across, either row-by-row or column-by-column. Because there is the potential for large spatial variations in roughness we can localize this measure we employ a sliding window technique. Take a w -row wide window, and slide it along a column and perform level set persistence. At the center of each window, the sum of the bars gives a local directional roughness score. Likewise, along rows, slide a window that is w -columns wide. w should be large enough to capture features of interest, but small enough to still give a local measure. This technique is applied to airborne light detection and ranging (LIDAR) estimates of the surface elevation of the snowpack of a snowfield in Niwot, CO. Small-scale features con-

tributing to surface roughness are captured in the bar lengths of the persistence diagrams while large variations in the underlying terrain affects the birth time of features. If we use bar length as a metric, then this disregards the terrain and focuses in on surface roughness.

There are some metrics for roughness that do not incorporate a directional component. For a persistence version of such a measure we use a two-dimensional cubical complex to compute persistence, again on subdomain or patch that will move to every spatial location possible in increments of a specified shift. The results of such a measure are shown in Fig. 4.3 for a larger patch size of 100×100 and in Fig. 4.5 for a small patch size of 20×20 . The second figure demonstrate a few other methods, besides the bar sum to summarize the information contained in the persistence diagrams for each patch.

A similar idea is used in 2D as well, sliding a smaller patch around the whole surface, and associating the barsum to the pixel at the center of the patch as a roughness measure. Window size is chosen so that a number of local peaks (which contribute to small scale roughness), but is small enough that it is not measuring large terrain-related peaks. Using this method, we can detect regions that are more rough due to surface morphologies. Future steps for this project involve a comparison between this method and other common methods.

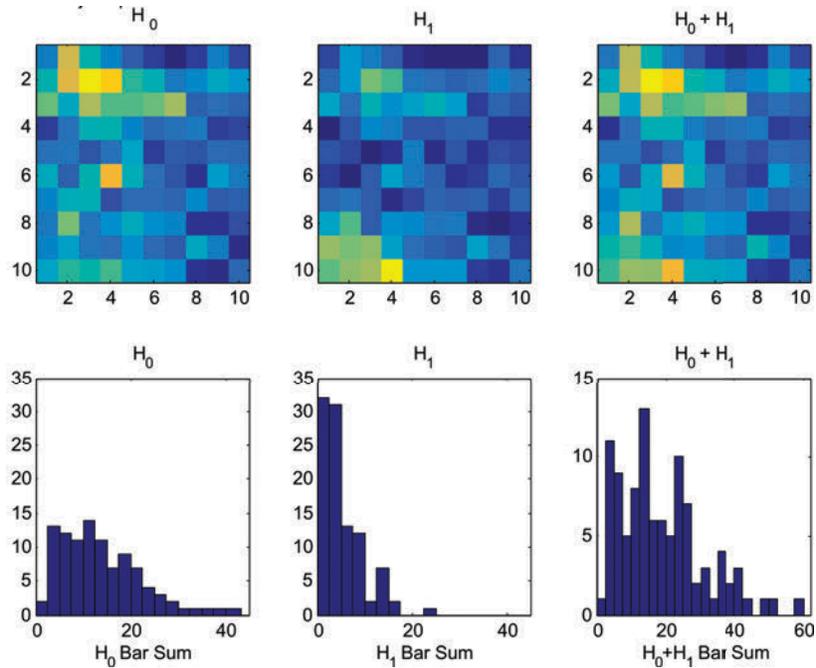


Figure 4.2: Dividing the domain into disjoint patches that are 100×100 m, one can compute a localized measure of roughness using the bar sum. At this scale, the larger topographical features do not contribute. Below a histogram of bar sums is shown for each homological dimension and for the combination.

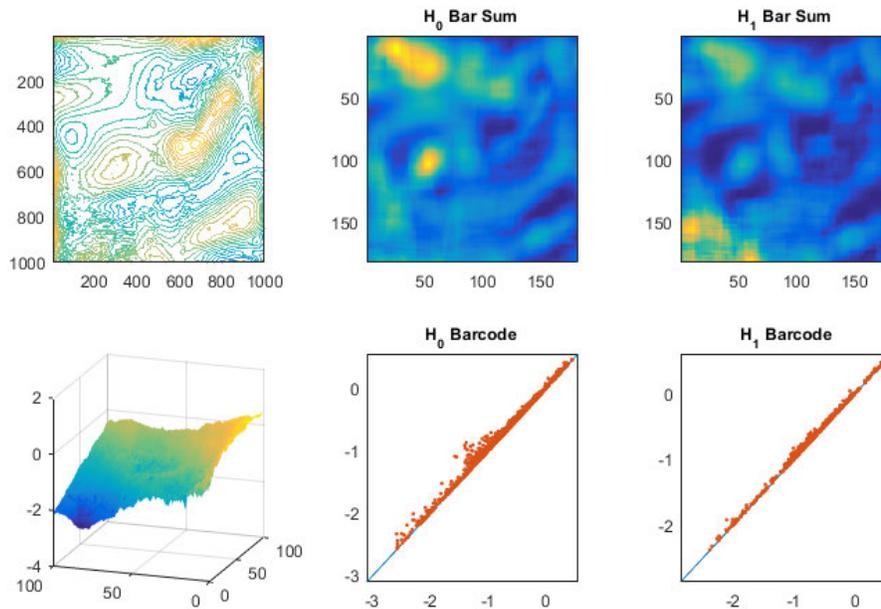


Figure 4.3: Lidar data of the entire region is shown as a contour plot in upper left. PH was computed using a 2D sublevel set filtration on patches of size 100×100 , with a shift of 5. For reference, the entire dataset is on a 1000×1000 spatial grid. The second row shows an example of a single patch with a high score, the indexing (matching the heat map of the bar lengths) is $i = 10, j = 26$, this corresponds to a patch at a similar location in the contour plot. H_0 and H_1 persistence diagrams are plotted as well.

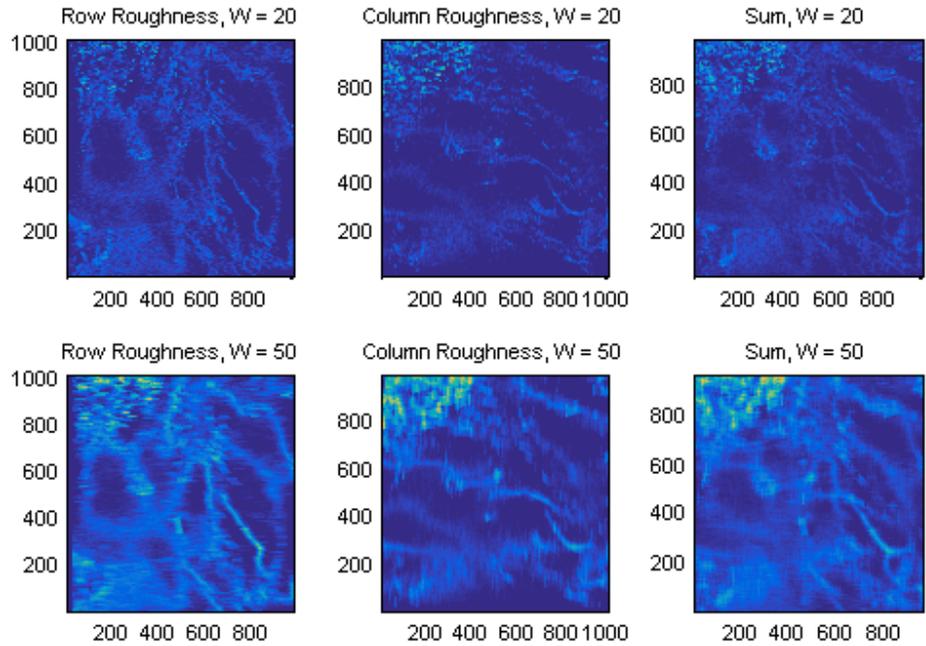


Figure 4.4: 1D sliding windows shown for window sizes $w = 20$ and $w = 50$. Row and column bar sums for each window are plotted at the spatial center of the window.

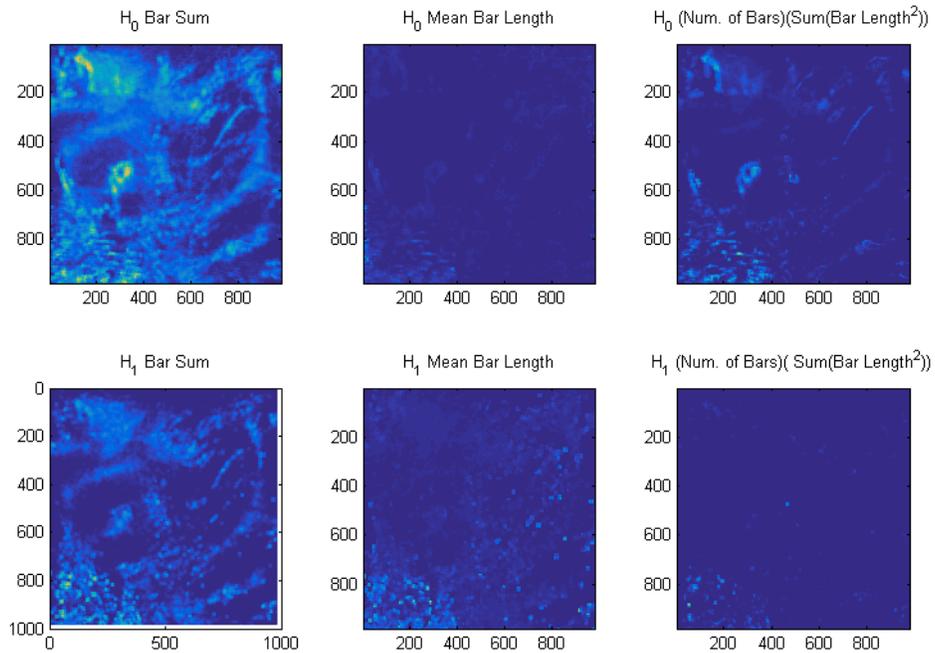


Figure 4.5: Several different methods of summarizing the persistence diagram for a patch size of 20×20 and a shift of one.

4.3 Pyramidal Surface Structures

Semiconductor surfaces become amorphous when subjected to ion bombardment, which can lead to periodic ripple patterns or hexagonally arranged nanodots. This is due to an interplay of preferential sputtering yield, momentum transfer and atomic currents on the surface [6]. The resulting model is an anisotropic Kuramoto-Sivashinsky-type equation, the Bradley-Shipman equations [6]. Ou et al. observed the novel formation of regular patterns of crystalline structures when elemental Ge(100) is irradiated at normal incidence and at elevated temperatures (above the recrystallization temperature) [89]. The result is highly-ordered arrays of four-sided inverted pyramids. They suggest that this is due to a reverse epitaxial growth mechanism. Hashmi et al. [53] extend the theory for the formation of these nanopylramids with fourfold rotational symmetry, formed on crystalline binary materials. An Ehrlich-Schwoebel (ES) barrier, which creates uphill atomic currents on the surface of the crystal, is essential for the formation of this pattern. The temporal evolution of the surface height is given in [89] by

$$\frac{\partial u}{\partial t} = -\nabla^2 u + \kappa \nabla^2 \nabla^2 u - \sigma \nabla^2 (\nabla^2 u)^2 + \delta (\partial_x u_x^3 + \partial_y u_y^3) \quad (4.1)$$

The second term, $\sigma \nabla^2 (\nabla^2 u)^2$ is called the “conserved Kadar-Parisi-Zhang” term is a non-linear current and is known to break up-down symmetry [89]. For $\sigma=0$, the the solution is invariant under the transformation $u \rightarrow -u$, and so we should not expect pyramids or inverted pyramids to dominate. The solutions coarsen with time, that is, the total number of pyramids will decrease in time. Ou et al. has claimed that for negative values of σ , the surface is dominated by “pits” or inverted pyramids and for positive values of σ , the surface is dominated by “mounds” [89]. This claim was made by inspection of the surface. We seek to quantify the difference in structure, with the aim of applying this statistic to laboratory experiments of irradiated Germanium.

We numerically simulate the solutions to equation 4.1, varying σ while holding all other parameters constant. We use a Fourier spectral method with periodic boundary conditions on a 256×256 spatial grid, with a fourth-order exponential time differencing Runge-Kutta method

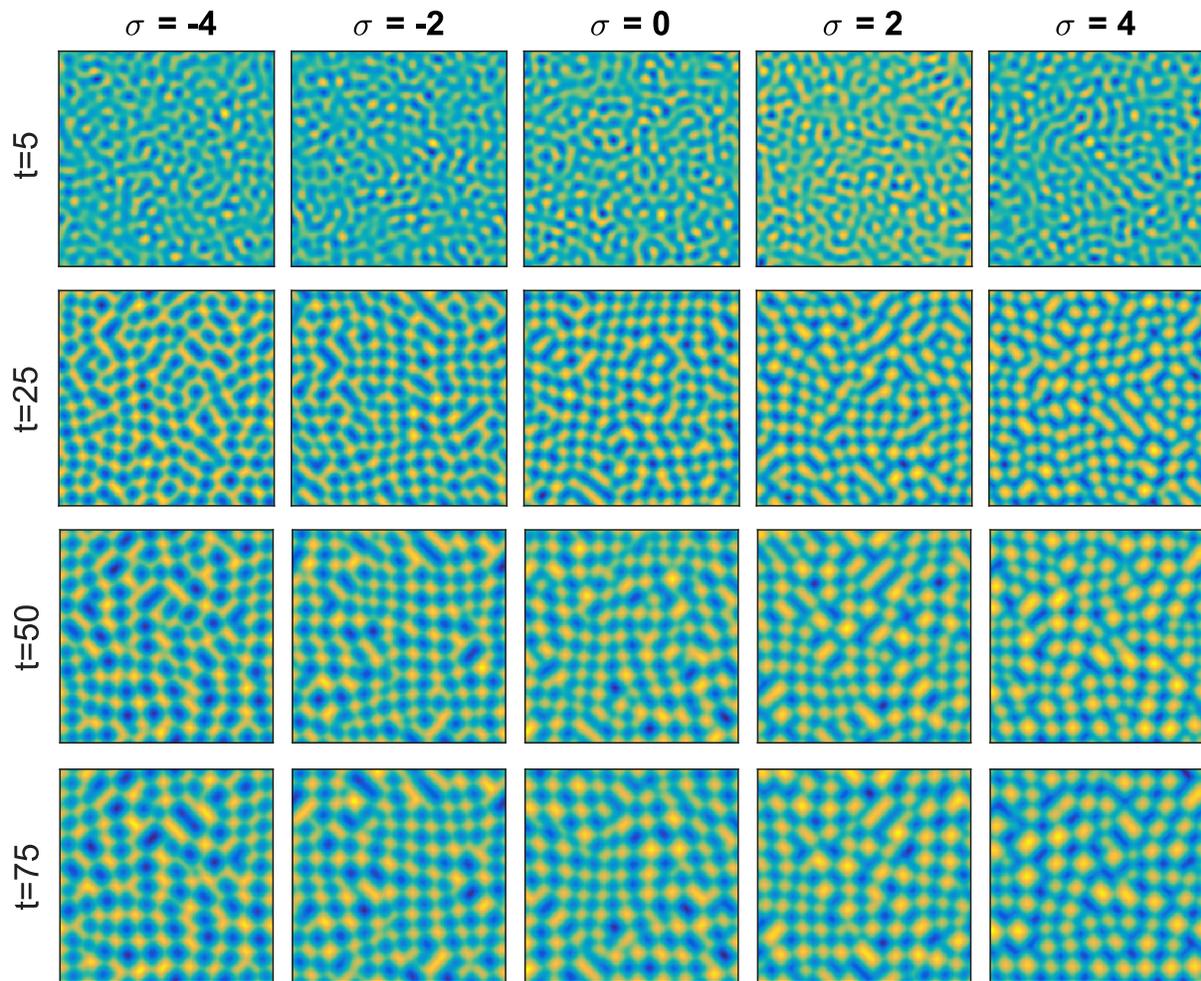


Figure 4.6: Plots of surface $u(x, y, \cdot)$ resulting from numerical simulations of equation 4.1. Each column corresponds to a parameter value, each row corresponds to a time step. We show a sampling of parameter values, our entire data set consists of 11 classes, for $\sigma = -5, -4, \dots, 4, 5$.

for the time stepping [16, 17]. Each trial is computed from low-amplitude white noise initial conditions. We compute 30 trials at each parameter. Following the parameters given in [89], which qualitatively align with experimental data, we let $\delta = 25$ and $\kappa = 4$. σ is varied from -5 to 5 by 1.

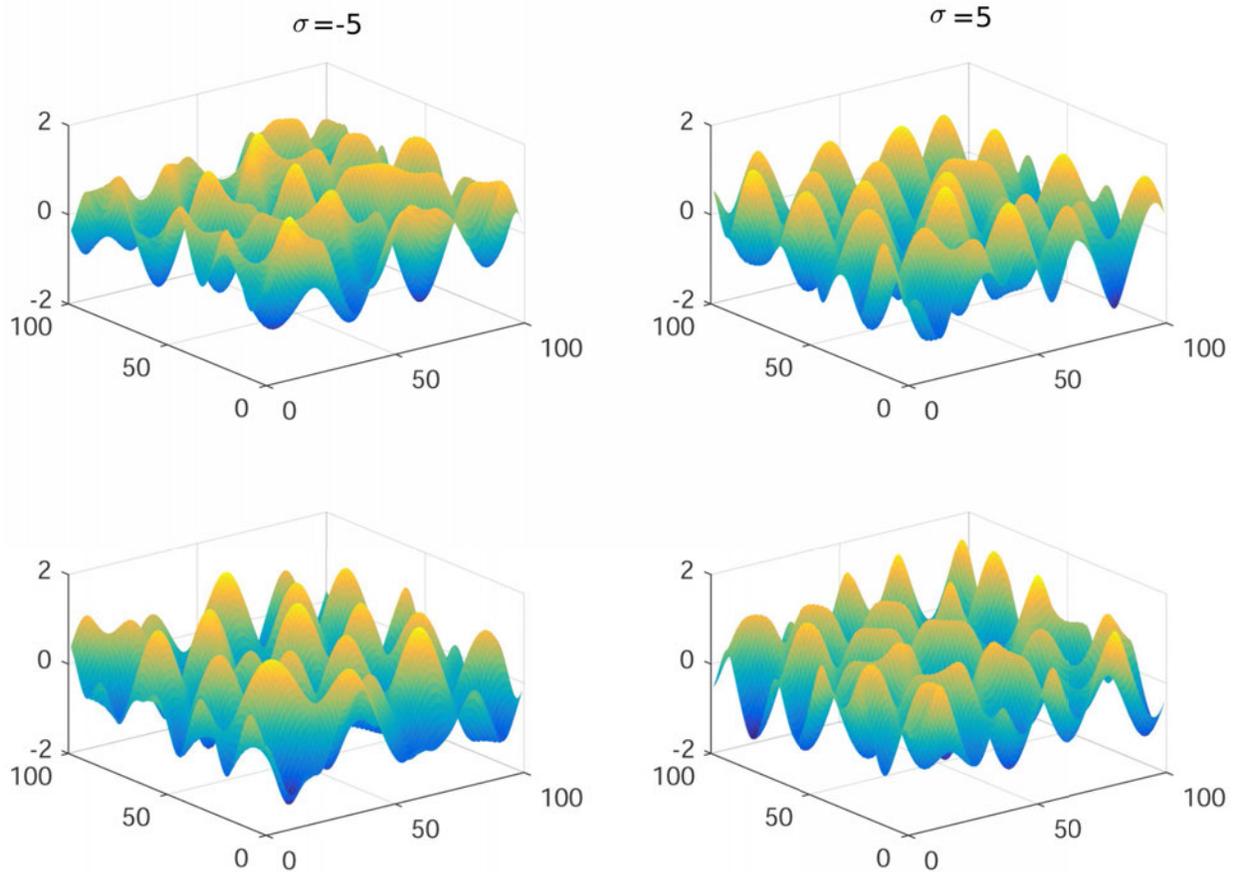


Figure 4.7: Plots of the surfaces $u(x, y, \cdot)$ that demonstrate the inverted pyramidal and pyramidal structure that forms when Ge is irradiated at elevated temperatures. Two perspectives are plotted to visually highlight the difference between well-formed pyramids and well-formed inverted pyramids. The first column is $\sigma = -5$ and the second column is $\sigma = 5$. The top row shows $u(x, y, \cdot)$ and the bottom row shows $-u(x, y, \cdot)$. One can see that in this example, $\sigma = 5$ and the inverted pyramidal structure is more prominent and for $\sigma = -5$ the pyramid structure is more prominent. We will quantify this difference.

We use cubical complexes to compute persistent homology of sublevel sets. Using a sublevel set filtration, for lower thresholds, the valleys will appear as individual connected components that merge as the threshold rises. For higher thresholds, the pyramids will appear as holes in

the sublevel set. Therefore, roughly speaking, the H_0 bars roughly correspond to the inverted pyramids and H_1 bars roughly corresponds to pyramids. The total length of the H_0 bars then gives a coarse notion of the prominence of inverted pyramids.

By duality, we note that there is a correspondence between the H_0 bars of a superlevel set filtration and the H_1 bars of a sublevel set filtration [13]. However, in building the cubical complex, we use an open topology for the cubical complex of the sublevel set and a closed topology for the cubical complex of the superlevel set [34]. This means that the H_0 bars corresponding to valleys are computed in a different way than the H_1 bars corresponding to the (in fact in this formulation we consider 8-neighbor connectivity for computing the H_1 features, so “holes” that correspond to peaks will result in a longer bar than the bar in H_0 corresponding to the same feature of the inverted surface. This means that using the mean bar length for H_0 and H_1 will always skew a little bit positive. For a fair comparison on the extent of the peaks and valleys, we consider H_0 of the sublevel set filtration of the surface $u(x, y, \cdot)$ (which will give an indication of the extent of the valleys) and H_0 of the sublevel set filtration of the inverted surface, $-u(x, y, \cdot)$. We compare and normalize in the following way:

$$P = \frac{\text{mean}(H_0(-u)) - \text{mean}(H_0(u))}{\text{mean}(H_0(-u)) + \text{mean}(H_0(u))}$$

where $\text{mean}(H_0(u))$ is the mean H_0 bar length of the persistence diagram computed on u . A positive P indicates there are more prominent peaks, a negative number indicates that there are more prominent valleys. We compute the pyramid score for each trial at each time step so that we can track how the pattern evolves over time, plotted in Fig. 4.8. Even as the pattern coarsens, this statistic remains relatively constant during the evolution of the pattern.

This statistic quantifies visual assessment; pyramids appear for positive values of σ and become more prominent and pronounced as σ increases. Its performance on simulated data indicates not only that quantifies the degree to which the pattern is dominated by pyramids or inverted pyramids, but also is promising for distinguishing and comparing patterns of laboratory experiments.

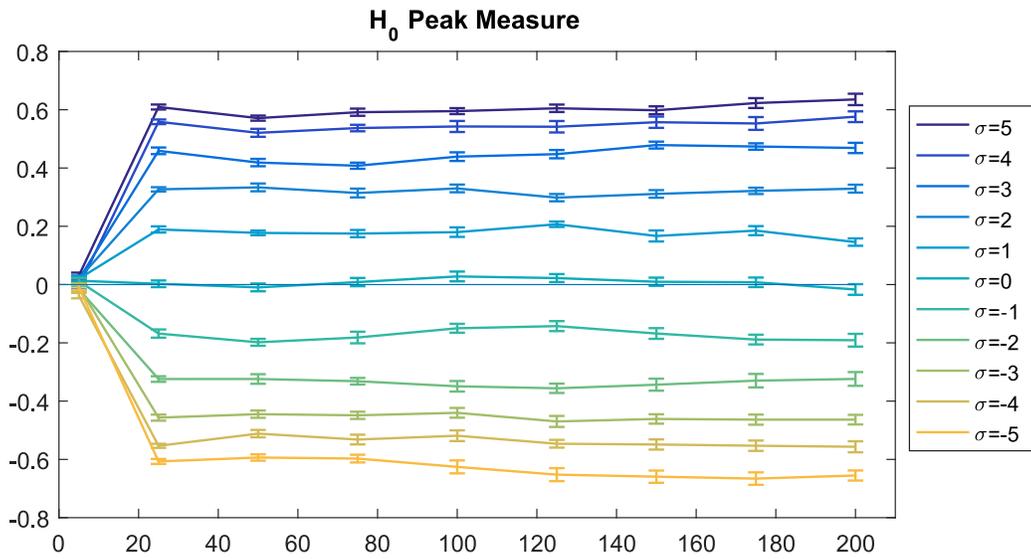


Figure 4.8: Mean and variance of the pyramid peak score over all trials, plotted as the system evolves in time. The earliest time plotted, $t=5$, is early enough in the pattern evolution that neither clear peaks nor clear valleys have appeared. See the top row of Figure 4.6.

Chapter 5

Topological Measure of Order

5.1 Lattice Patterns

Lattice patterns can arise from various natural processes and laboratory experiments, the most common of which are hexagonal lattices. Some of the most common examples are found in Raleigh-Bénard convection experiments [45], the Rosenzweig instability in ferrofluids [49], and nanoscale structures formed by bombarding a binary material by a broad ion beam [5–7, 105]. While these are well-ordered systems, they rarely produce perfect hexagonal lattice patterns. These patterns often contain defects. Penta-hepta pairs, where one point will have 5 nearest-neighbors and one of its neighboring points will have 7 nearest-neighbors, rather than 6, as is the case for a perfect lattice. Grain boundaries which occur where regions of different lattice orientations come together also cause defects.

The near hexagonal patterns of nanodots produced by ion bombardment form often contain many such defect. Quantifying such defects, and studying their evolution as the pattern evolves and potentially resolves to a well ordered systems becomes important. There are a number of methods for quantitatively gauging order. For example, computing the width of the lowest-order peak in the Fourier transform of the surface can indicate order. A narrow Fourier peak indicates a highly ordered structure. However, the Fourier peaks may not be separable from the background or each other. Another common technique is considering how the auto-correlation function decays with distance. If the decay is exponential for some distance, then the characteristic length scale of the decay gives an estimate for the range over which order exists. However, there may not be a region of exponential decay. Böttger et al [5] computed the characteristic length for a series of imperfectly ordered hexagonal arrays of nanodots and showed that it remained nearly constant even as the order increased.

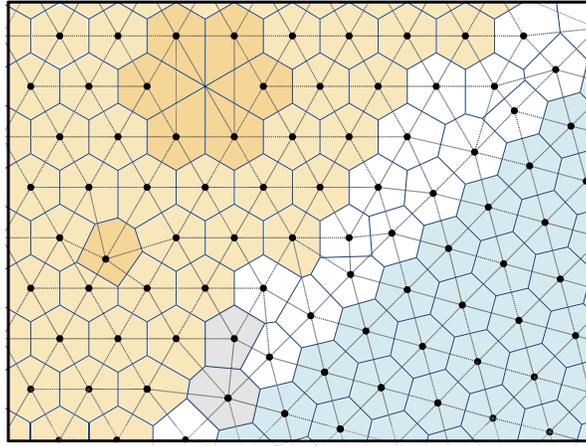


Figure 5.1: Examples of possible defects present in near hexagonal patterns. The gold and blue regions are nearly perfect hexagonal lattices with different orientations. They meet along a grain boundary, which is left white. Along the grain boundary, there is a pent-hepta defect in gray. Shaded darker gold is a defect formed by a displaced lattice point and one formed by a missing lattice point. The dotted gray lines indicate the Delaunay triangulation and navy the Voronoi cells for this lattice.

A measure of order based on persistence will prove to be a useful tool to complement the current suite of tools in this context to detect defects and quantify order of nearly perfect and noisy lattices. This measure was introduced in [91] and developed in [82]. In physics persistence has been used to study the hierarchical structure in glasses [85], to characterize the structure of granular media and the force networks in them [3, 64, 65], and to study fluid flow [66].

5.2 Measures of Order

Defects in lattice structures occur in a number of ways. Figure 5.1 has two regions of nearly perfect hexagonal regions with different lattice orientations, shaded gold and blue. The white region in between the two highly ordered regions is the grain boundary. There are two defects shaded darker in the gold region. In the lower left corner, there is a point that is displaced from a perfect lattice. Above is a single lattice point removed. Shaded in gray along the grain boundary is a penta-hepta defect (there are actually several). A good measure of order should be able to detect these types of defects. We describe in Section 5.2.1 spectral (that is, using the Fourier transform), geometric-combinatorial (using the Delaunay triangulation) and topological (using

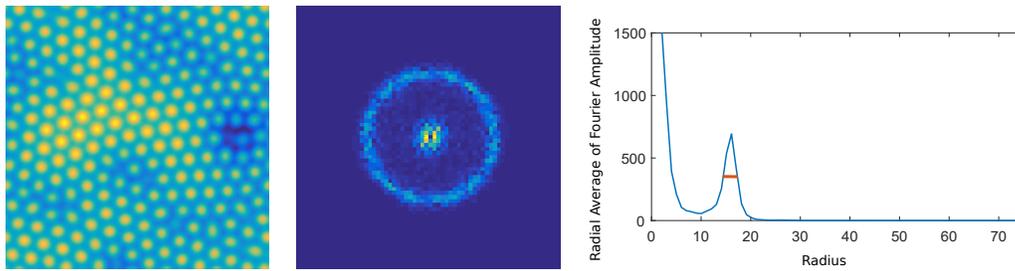


Figure 5.2: Left, an example of an ion bombarded binary surface exhibiting hexagonal order with defects. Center, the Fourier transform of the surface. Right, the radial average of the Fourier transforms. The FWHM measure is marked in orange.

persistent homology) tools for measuring characteristics of sets of points in the plane. In Section 5.3, we use each of these tools to define measures for order for nearly hexagonal lattices.

5.2.1 Spectral, Geometric-Combinatorial, and Topological Tools for Describing Lattice Patterns

The Fourier Width at Half Maximum

Measures of order may be based on the Fourier transform of a pattern. The full-width-at-half-max (FWHM) of the first Fourier peak is a spectral method computed from the Fourier transform of the lattice. The spatial domain is discretized and a delta function is centered on each lattice point. The 2-dimensional Fourier transform is computed and shifted so that the zero-frequency component is centered. The radial average of the Fourier transform is computed from the center by averaging all pixels that are distance $r - \epsilon$ to $r + \epsilon$ from the center. r values are chosen by taking the distance to the center of each pixel along the horizontal line through the center. ϵ is the distance between two pixels on this axis. FWHM is computed by finding the width of the first Fourier peak at half of the height of the peak in radially averaged spectrum. This is done using a linear interpolation between two points when half of the maximum value falls is between known values of our radial average. A narrow first Fourier peak indicates a near perfect lattice.

Autocorrelation Function

Pichler et al. introduce the use of a local autocorrelation function in [93]. For a discrete set of 2D data, size $N \times M$, the discrete 2-dimensional autocorrelation function (ACF) is given by

$$2D - ACF_{discrete}(j, l) = \frac{1}{(N-j)(M-l)} \sum_{i=1, k=1}^{N-j, M-l} (a_{i,k} - \bar{a})(a_{i+j, k+l} - \bar{a})$$

where j and l are shifts, the sum is over all points that overlap under the given shift j and l . The maximum amplitude is always at the center and is symmetric in antiparallel directions.

For a hexagonal lattice, the 2D-ACF is periodic. The period is the average inter-particle distance [93]. (For nonperiodic data, the ACF decays following a Gaussian distribution.) For an imperfect lattice, there is a decay in the peak heights of the ACF. The ACF is fitted to a Gaussian envelope and the rate of decay is a measure of the “imperfectness” of the lattice structure. The decay strongly depends on the type of imperfections present. Fitting to a linear combination of two Gaussians:

$$ACF_{fit}(r) = A_{\infty} + f_0 e^{-\frac{9}{2}(r)^2} + f e^{-\frac{1}{2}(r/\sigma)^2}$$

where A_{∞} is the asymptotic peak level. This gives ordering metrics that capture both short range and long range ordering. These measure are very sensitive to small perturbations in long range or short range ordering. This method considers line profiles along the main axis of the 2D-ACF. To approach data that displays anisotropy, the peaks in the 2D-ACF are weighted by $1/r$ (to deal with the fact that peaks farther away from the center are over represented). These points are plotted against their distance to the center. The data is fit and the average square residual is computed. This gives a characterization of the anisotropy in the data.

Pourfard et al. [94] generalize the autocorrelation method of Pichler et al. to a global autocorrelation method. Peaks in the autocorrelation function will appear wherever a texton (building block of the texture) is repeated. For a perfectly ordered pattern, the peaks will all have the same size. Less ordered patterns will display a decay in the peak sizes. The slope of the line between peaks can then be used as a measure of order. A higher slope indicates less order. They

do note that in order to perform a fair comparison, background noise must be normalized. This measure is capable of detecting different types of lattice imperfections such as a defect or regional distortion [93] and enables quantitative comparison of different lattice structure. However, it is robust to small defects.

Delaunay Triangulation

Given a set P of points in the plane, the plane may be decomposed into a Voronoi tessellation, namely the union of Voronoi polygons V_p for $p \in P$. The Voronoi polygon V_p is the set of all points in the plane that are closer to p than to any other point in P . That is, $V_p = \{x \in \mathbb{R}^2 : |x - p| < |x - q| \text{ for all } q \in P, q \neq p\}$. The Delaunay triangulation of the set P is a graph (vertices and edges) formed as the dual to the Voronoi tessellation in the following way: The vertices of the Delaunay triangulation are the points in P , and there is an edge connecting points p and q in P if the Voronoi polygons V_p and V_q share a side. The Delaunay triangulation and Voronoi tessellation are shown in Figure 5.1. For a perfect hexagonal lattice, every lattice point will be connected to six neighbors (equivalently, the Voronoi polygons are hexagons).

Persistent Homology

We can also use persistent homology to characterize the both the severity of individual defects in nearly hexagonal, planar lattices and a way to quantify the overall degree of order with a single number. In the context of a lattice, we use a Vietoris-Rips filtration to compute persistence. Fig. 5.3 shows the H_0 and H_1 persistence diagrams for nearly hexagonal data. Notice the variance in the lengths of the intervals in the H_0 barcode. For a perfect lattice, almost all connected components collapse into a single connected component at the same connectivity parameter r_c , the nearest-neighbor distance in the hexagonal lattice without defects. However, the defects result in some components collapsing at values of r less than r_c . For a perfect hexagonal lattice, points will connect and be filled in with a 2-simplex simultaneously. This means that there is no H_1 signal. Any H_1 bars indicate the presence of defects. The more severe the de-

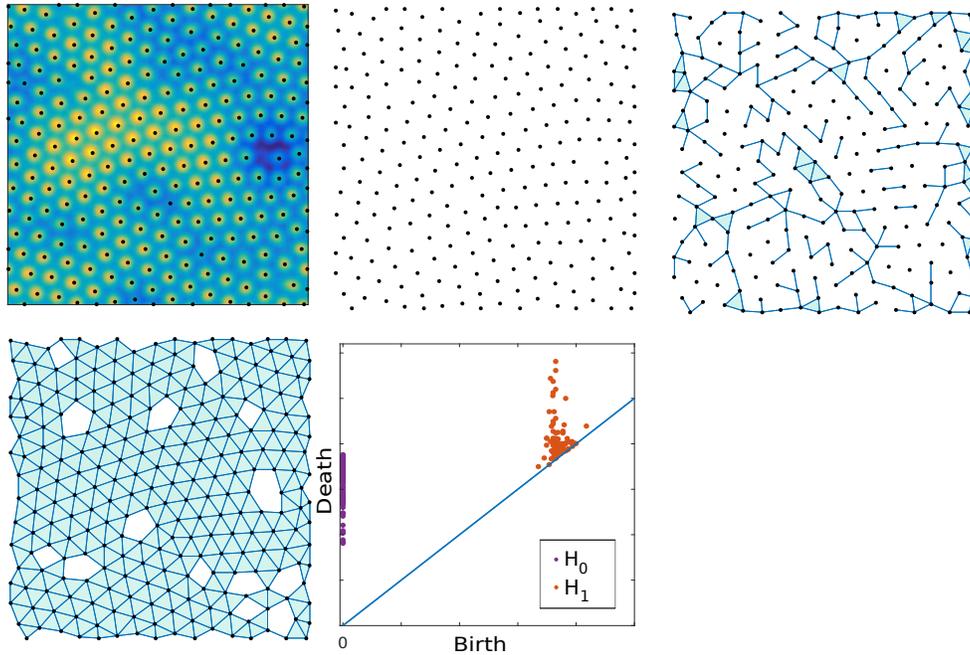


Figure 5.3: The first panel shows the solution with the peaks indicated with black dots. The third and fourth panels show two examples of VR complexes in the filtration. Notice that larger defects appear as holes when the radius is increased passed the average inter-dot distance. The resulting persistence diagram is shown in panel 5. For a perfect hexagonal lattice, one expects one H_0 bar with multiplicity and no H_1 signal. Defects appear in the persistence diagram as deviations from such a persistence diagram.

fect, the larger the bar length. The fourth panel in 5.3 shows a threshold value at which several defects in the lattice appear as holes.

5.3 Characterizations of Order

Having introduced the Fourier width at half max, Delaunay triangulations, and persistent homology, we are prepared to define and compare several measures of order in nearly hexagonal lattices. We motivate and define these measures through the consideration of Bravais lattices in Section 5.3.1, and then test the measures on perturbations of Bravais lattices in Sections 5.3.3 and 5.3.4.

5.3.1 Bravais Lattices

A Bravais lattice in \mathbb{R}^2 is the integer linear combination $\mathbb{L} = \{z_1 \vec{v}_1 + z_2 \vec{v}_2 : z_i \in \mathbb{Z}\}$ of a basis $\{\vec{v}_1, \vec{v}_2\}$ for \mathbb{R}^2 . In this section, we apply the N^3 and PH measures of hexagonal order to Bravais lattices. An ideal measure of hexagonal order should designate only a perfectly hexagonal Bravais lattice to be perfectly ordered.

Two Bravais lattices \mathbb{L}_1 and \mathbb{L}_2 are equivalent for our purposes if there exists an angle-preserving linear transformation that is a bijection between them. The shaded regions in Fig. 5.4 depict two representations of the entire set of equivalence classes of Bravais lattices, determined as follows: Given a Bravais lattice \mathbb{L} , choose a shortest vector $\vec{u}' \in \mathbb{L}$; that is,

$$|\vec{u}'| = \min_{\vec{v} \in \mathbb{L}} |\vec{v}|.$$

Choose also a vector $\vec{v}' \in \mathbb{L}$ that is a vector of minimal length not in the span of \vec{u}' ;

$$|\vec{v}'| = \min_{\vec{v} \in \mathbb{L} - \text{span}\{\vec{u}'\}} |\vec{v}|.$$

For any such vector \vec{v}' , $-\vec{v}'$ also satisfies the same condition. This allows us to choose \vec{v}' so that $|\vec{u}' - \vec{v}'| \leq |\vec{u}' + \vec{v}'|$, or equivalently so that the angle θ between \vec{u}' and \vec{v}' satisfies $0 < \theta \leq \frac{\pi}{2}$.

Now let T be the angle-preserving linear transformation of \mathbb{R}^2 such that $T(\vec{u}' - \vec{v}') = (1, 0)$ and $T(\vec{u}')$ lies in the first quadrant. T is a combination of scaling (by $1/|\vec{u}' - \vec{v}'|$), rotation, and reflection. Write $\vec{u} = T(\vec{u}') = (x, y)$, $\vec{v} = T(\vec{v}')$, and $\vec{w} = T(\vec{u}' - \vec{v}') = \vec{u} - \vec{v} = (1, 0)$. The choices of \vec{u}' , \vec{v}' and T imply that $0 < x, y$; $|\vec{u}| \leq |\vec{v}| \leq 1$; and $|\vec{u} - \vec{v}| \leq |\vec{u} + \vec{v}|$. These constraints determine a region R in \mathbb{R}^2 in which $\vec{u} = (x, y)$ may lie. R is the region bounded by the following curves:

1. $|\vec{u}| \leq 1$ implies that $x^2 + y^2 \leq 1$.
2. $|\vec{v}| \leq 1$ implies that $(x - 1)^2 + y^2 \leq 1$.
3. $|\vec{u}| \leq |\vec{v}|$ implies that $x \leq \frac{1}{2}$.
4. $|\vec{u} - \vec{v}| \leq |\vec{u} + \vec{v}|$ (equivalently, $0 \leq \vec{u} \cdot \vec{v}$) implies that $0 \leq x^2 - x + y^2$.

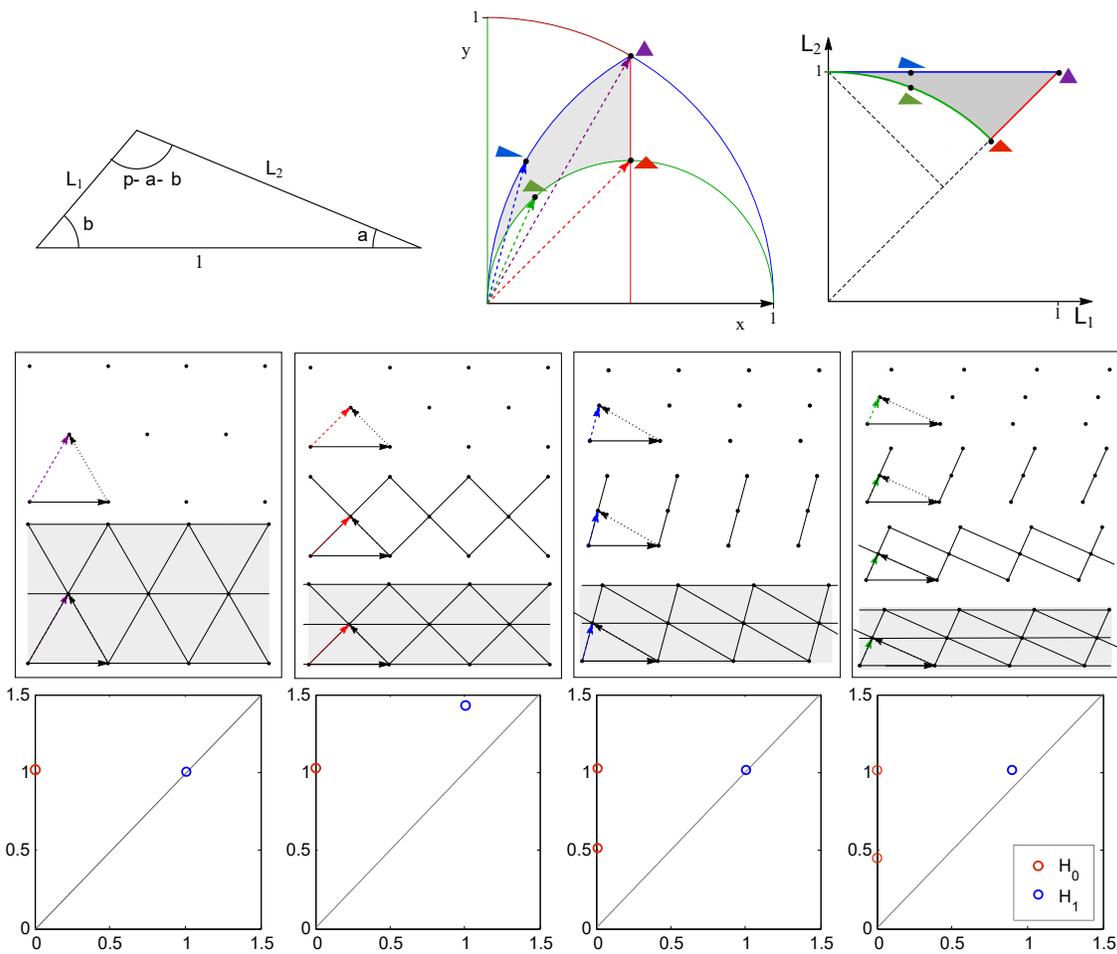


Figure 5.4: Top row: The region R parameterizing the space of inequivalent Bravais lattices in the x, y -plane (first column) and the L_1, L_2 -plane (third column), where as depicted in the second column, the side lengths of a fundamental triangle for the Bravais lattice are $L_1 \leq L_2 \leq 1$. Second row: Each column is an example of a Bravais lattices and the sequence of simplicial complexes formed for values of the connectivity parameter r noted on the diagrams. For each sample Bravais lattice, the vector $(1,0)$ is marked by a solid black arrow, the vector \vec{u} that lies in the region R is marked by a dashed colored arrow, and the vector \vec{v} (see text) is marked by a dashed black arrow. These vectors form the fundamental triangle for the lattice. For each sample lattice, the vector \vec{u} and the shape of the fundamental triangle is marked on the depiction of the region R in the first column of the first row. The H_0 and H_1 persistence diagrams for the sample Bravais lattices are shown in the third row.

The upper left panel of Fig. 5.4 depicts the region R in the (x, y) -plane. The shaded region is R ; a choice of $\vec{u} = (x, y)$ outside of this region determines a lattice that may be transformed to one in R by an angle-preserving transformation. Colors code for curves on which $\vec{u} \cdot \vec{v} = 0$ (green), $|\vec{u}| = |\vec{v}|$ or $|\vec{u}| = |\vec{w}|$ (red), and $|\vec{v}| = |\vec{w}|$ (blue).

Examples of Bravais lattices and the triangle Δ formed by the vector triad \vec{u} (colored, dotted), \vec{v} (black, dotted), and \vec{w} (black, solid line) are shown in the second row of Fig. 5.4. The triangle Δ is crucial to the computation of the measures of order. Its side lengths are $L_1 = |\vec{u}|$, $L_2 = |\vec{v}|$, and $|\vec{w}| = 1$, as depicted in the upper central panel of Fig. 5.4. It is helpful to also visualize the region R in terms of the side lengths L_1 and L_2 of Δ , as depicted in the upper right panel of Fig. 5.4.

For each example Bravais lattice in Fig. 5.4, the position of the corresponding fundamental triangle is marked in the plots of the region R . These examples include a hexagonal lattice ($L_1 = L_2 = 1$; purple), a square lattice ($L_1 = L_2 < 1$; red), a lattice for which $L_1 < L_2 = 1$ (blue), and a lattice for which $\vec{u} \cdot \vec{v} = 0$ and $L_1 < L_2 < 1$ (green).

Measures of hexagonal order using the Delaunay triangulation

For a perfect hexagonal lattice, each polygon in the Voronoi tessellation is a hexagon, so the number of sides of each Voronoi polygon is 6. Equivalently, the degree of each vertex in the Delaunay triangulation is 6. For the square or rectangular Bravais lattice, the number of nearest neighbors to any point is 4, and defects in a lattice can alter the degree of vertices in the Delaunay triangulation, as shown in Fig. 5.5. The N^3 measure is computed by taking the variance of $N_i - 6$, where N_i is the number of nearest neighbor of the i th lattice point. This will give a N^3 score of 0 for a perfect hexagonal lattice. The N^3 measure is, however, a very crude measure of hexagonal order, since the N^3 of any lattice that is not rectangular (such as a square lattice) is 0.

The DT measure of order is a statistic on the lengths of the edges in a Delaunay triangulation. For perfect hexagonal ordering, each triangle in the triangulation is equilateral. Compute the dominant edge length, L_{dom} , for a lattice by averaging over all edge lengths appearing in the triangulation. Assign to each triangle a score: $L_{i,dev} = \sum_{k=1}^3 |L_{i,k} - L_{dom}|$ where $L_{i,k}$ is the edge

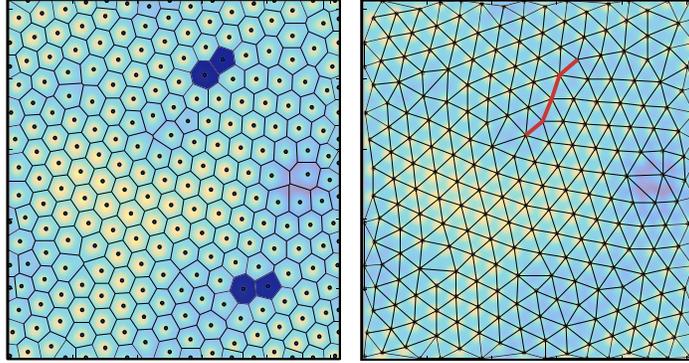


Figure 5.5: (Left) Voronoi cells of the peaks from ion data. Shaded are two penta-hept defect pairs. (Right) The Delaunay triangulation on the same set of peaks. Defects appear as a distortion of equilateral triangles in the triangulation.

length of the k th edge of the i th triangle. The DT measure is the mean of $L_{i,dev}$. For a perfect hexagonal lattice, the DT measure will be 0. This is a simplification of a measure used in Mátéfi-Tempfli et al. [73] and has the advantage of being highly computable and effective in discrimination between examples of varying degrees. However, it is less sensitive for long-range order, in highly ordered structures [73] and does not take into account orientation of regions [94].

In the Delaunay triangulation, triangles appearing along the edges of the domain can be highly distorted. To avoid edge effects from such triangles, tile the area surrounding the lattice with copies of the same lattice, compute the triangulation of the tiled domain and only retain edges that appear both in the triangulation of the original domain and the tiled domain.

For a Bravais lattice, the DT measure of order is $L_1^2 + L_2^2 - L_1L_2 - L_1 - L_2 + 1$. We show in the first and third panels of Fig. 5.6 the DT measure of order as a function of Bravais lattice in the region R . Note that the DT measure is an insensitive measure of hexagonal order for lattices with $L_1 = L_2$ (marked by the red curves in the figures).

The PH Measures of Order:

The first step in using PH to measure order of Bravais lattices is to determine the nested sequence of simplicial complexes formed as the connectivity parameter r increases. For the four example Bravais lattices of Fig. 5.4, we show the entire nested sequence of simplicial complexes

(second row), and the corresponding H_0 (third row) and H_1 (fourth row) persistence diagrams. The generic case $0 < L_1 < L_2 < 1$ for which there are four simplicial complexes in the nested sequence, illustrated by the fourth column of Fig. 5.4, applies to all lattices represented by the interior of the region R as well as those on the (green) curve given by $\vec{u} \cdot \vec{v} = 0$.

Recall that the H_0 persistence diagram is computed by observing the changes in the numbers of components in the simplicial complex as r increases. The number of components is equal to the number of points at $r = 0$, decreases at $r = L_1$, and decreases to 1 at $r = L_2$. In the H_0 persistence diagram, there are therefore an equal number of bars of lengths L_1 and L_2 . For the hexagonal lattice, $L_1 = L_2$, so there is only one barlength. The variance of the barlengths in the H_0 persistence diagram, is therefore 0 for a perfect hexagonal lattice. This variance, $\text{var}(H_0)$ is our first topological measure of hexagonal order. For a Bravais lattice, the H_0 variance is

$$\text{var}(H_0) = \frac{1}{4}(L_2 - L_1)^2.$$

The H_1 persistence diagram is computed by observing changes in the numbers of topological holes in the simplicial complex as r increases. For a Bravais lattice, topological holes form at $r = L_2$ and are filled in by faces at $r = 1$. Since $L_2 = 1$ for a perfect hexagonal lattice, the H_1 persistence diagram for such a lattice is empty (devoid of bars). We define a second topological measure of order as the sum of all the lengths of the bars in the H_1 persistence diagram and call this the H_1 sum, ΣH_1 . For a Bravais lattice, the H_1 sum is therefore

$$\Sigma H_1 = 1 - L_2$$

times the number of holes.

Fig. 5.7 shows, as functions of position in the region R , $\text{var}(H_0)$ (first column) and ΣH_1 (second column); darker shading indicates a larger value. $\text{var}(H_0)$ ranges from 0 along the entire line $L_1 = L_2$ (in red) which includes both the hexagonal and square lattices to $\frac{1}{4}$ as L_1 approaches 0

(and L_2 approaches 1). ΣH_1 ranges from 0 along the entire curve $L_2 = 1$ (in blue) to $(\sqrt{2} - 1)/\sqrt{2}$ at $L_1 = L_2 = 1/\sqrt{2}$.

Both $\text{var}(H_0)$ and ΣH_1 are equal to zero along 1-dimensional curves that intersect at exactly one point, namely the point representing the hexagonal lattice. Hence, although neither measure completely distinguishes the hexagonal lattice, a linear combination $c_1 \text{var}(H_0) + c_2 \Sigma H_1$ with positive constants c_1, c_2 is equal to zero only for the hexagonal lattice. Choosing $c_1 = 2$ and $c_2 = 1/(2 - \sqrt{2})$ so that the maximum values of $c_1 \text{var}(H_0)$ and $c_2 \Sigma H_1$ in the region R are both equal to $\frac{1}{2}$, we plot in the third column of Fig. 5.7 the combined PH measure of order

$$CPH = 2\text{var}(H_0) + \frac{1}{2 - \sqrt{2}} \Sigma H_1.$$

For a Bravais lattice,

$$CPH = \frac{1}{2}(L_2 - L_1)^2 + \frac{1}{2\sqrt{2}}(1 - L_2).$$

The DT measure of order turns out to nearly also be a linear combination of $\text{var}H_0$ and ΣH_1 .

Indeed,

$$DT = \frac{9}{6}(4\text{var}(H_0) + (1 - L_1)\Sigma H_1).$$

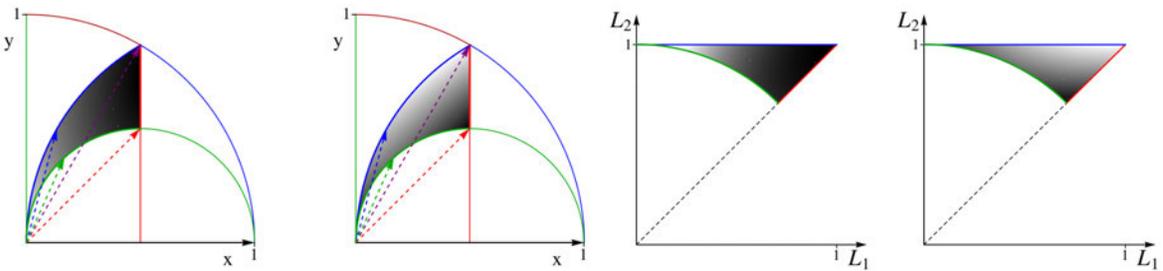


Figure 5.6: Gray-scale plots (with darker shading indicating larger values) of the DT (panels 1 and 3) and total PH (panels 2 and 4) measures of order within the region R parameterizing Bravais lattices, depicted in the x, y -plane (panels 1 and 2) and the L_1, L_2 -plane (panels 3 and 4).

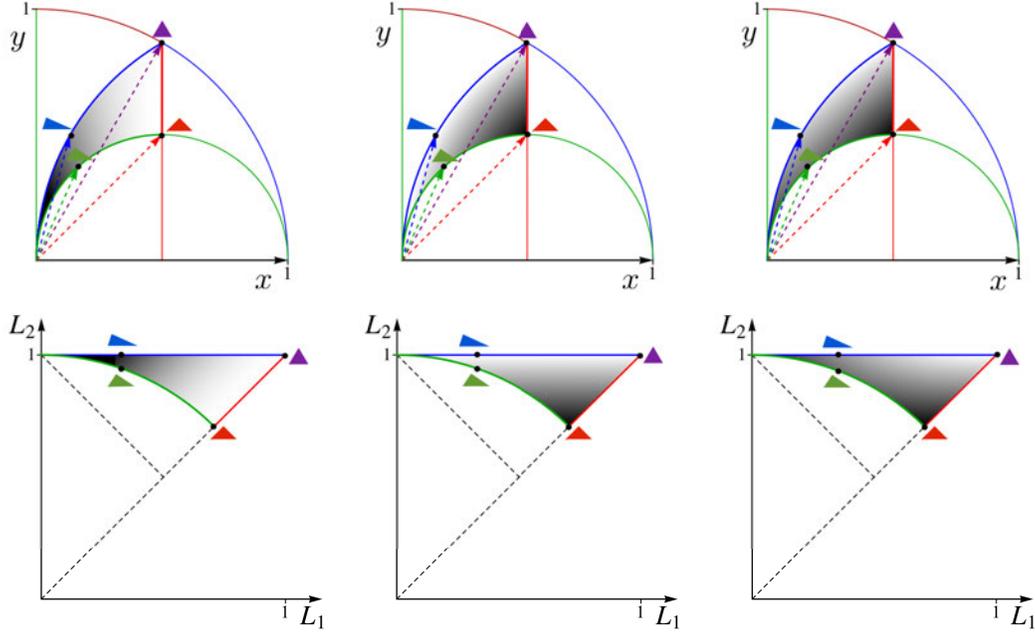


Figure 5.7: Gray-scale plots (with darker shading indicating larger values) of the PH measures of order within the region R parameterizing Bravais lattices, depicted in the x, y -plane (top row) and the L_1, L_2 -plane (bottom row). The measures $\text{var}(H_0)$, ΣH_1 , and $c_1 \text{var}(H_0) + c_2 \Sigma H_1 = 2\text{var}(H_0) + 1/(2 - \sqrt{2})\Sigma H_1$ are respectively graphed in the first, second, and third columns.

5.3.2 Variations on Filtrations

For the previous computations, a Vietoris-Rips filtration was used to create the filtration of which persistence was computed. This had a distinct advantage in the case where lattice points were near perfectly hexagonal. The signature of a perfectly hexagonal lattice is that $\text{var}(H_0)$ and ΣH_1 are both zero. Computing persistence using the Čech complex results in a slightly different ΣH_1 .

We recall the difference between the Čech complex and the Vietoris-Rips (VR) complex. Let X be a finite set of points in \mathbb{R}^n . We center balls $B_x(r)$ at each point x . To build the associated Čech complex, add a d -simplex when there is a common point of intersection of all d ($\epsilon/2$)-balls. To build the associated VR complex, add a d -simplex when $d+1$ ϵ -balls all have nonempty pairwise intersections.

In a Čech complex, the 2-simplex corresponding to the fundamental triangle will not be added to the filtration until the threshold value that corresponds with the radius of the circum-

Table 5.1: PH and PH statistics for the VR and Čech complexes on a Bravais lattice. Note that ΣH_1 is multiplied by n , the number of H_1 bars

Complex	H_0	H_1	$\text{var}(H_0)$	ΣH_1
VR	$[0, L_1], [0, L_2]$	$[L_2, 1]$	$\frac{1}{4}(L_2 - L_1)^2$	$(1 - L_2)n$
Čech	$[0, \frac{L_1}{2}], [0, \frac{L_2}{2}]$	$[\frac{L_2}{2}, \frac{L_1 L_2}{2y}]$	$\frac{1}{16}(L_2 - L_1)^2$	$\frac{nL_2(L_1 - y)}{2y}$

circle containing the three vertices of our fundamental triangle. This radius is given by $\frac{L_1 L_2}{2y}$, where (x, y) is the coordinate of the third vertex of the fundamental triangle.

Similar to Fig. 5.7, one can plot functions of position in the region R , $\text{var}(H_0)$. $\text{var}(H_0)$ for the Čech complex is the same as $\text{var}(H_0)$ for the Rips complex, scaled by $\frac{1}{4}$ due to convention of choosing the radius or diameter of a ball to index the threshold. ΣH_1 is 0 only for a square lattice and is maximized at $L_1 = L_2 = \sqrt{2}/2$. $\Sigma H_1=0$ therefore distinguished a square lattice.

Having used Bravais lattices to motivate definitions of spectral (Fourier Width at Half Max), combinatorial (N^3), geometric-combinatorial (DT), and topological ($\text{var}(H_0)$, ΣH_1 , CPH) measures of order, we now use these measures on perturbed Bravais lattices, returning to a Vietoris-Rips filtration.

5.3.3 Bravais lattices perturbed at one point

To understand the sensitivity of each measure to small perturbations in the lattice, we start by perturbing a single point in a perfect hexagonal lattice. For each configuration, we compute the various measures of order and at the location of the perturbed center, colored by the value of the measure of order. This is shown in Fig. 5.8. The point at the center of the hexagon represents a perfect, unperturbed lattice. The vertices of the hexagonal region are the nearest unperturbed lattice points. From this, we can see a large region in which the N^3 is not sensitive to a perturbation of the center point. In fact, in the whole center region, the value of the N^3 is zero since the center is not perturbed enough to change the configuration of the Delaunay triangulation until the point enters the semi-circular regions along the edges. The DT measure displays a small discontinuity in values along the same contour for the same reason. Notably, there are much smaller regions where $\text{var}(H_0)$ and ΣH_1 are less sensitive. Taking a linear com-

bination of the two measures provides better sensitivity to small perturbations of a single point in the lattice.

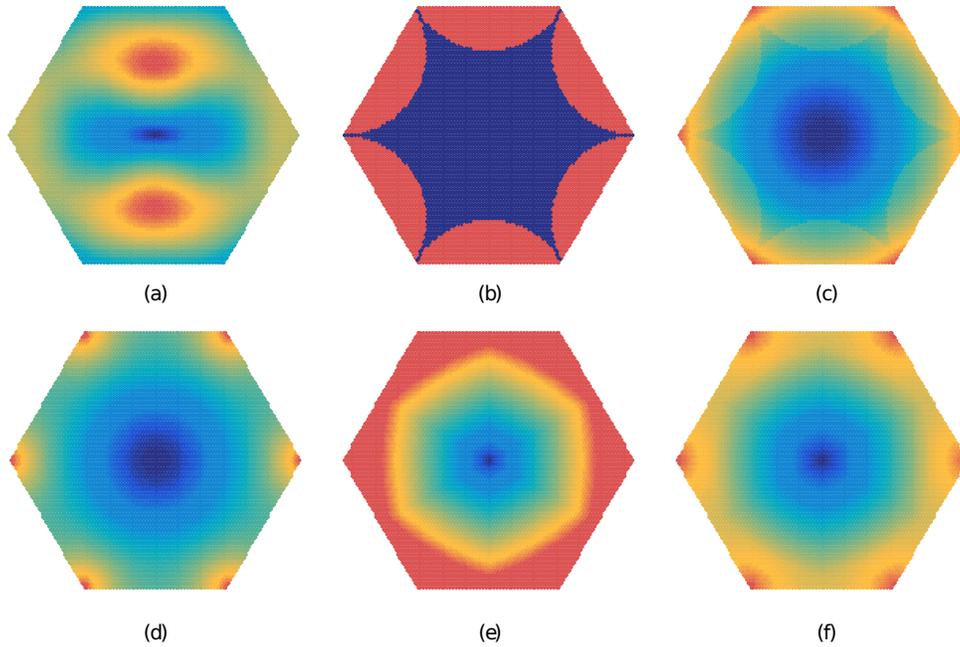


Figure 5.8: Measures of order for a perfect hexagonal lattice with the center point perturbed: (a) FWHM, (b) DT, (c) N^3 , (d) $\text{var}(H_0)$, (e) ΣH_1 , and (f) CPH. The location of the plotted point gives the location of the perturbed center point and the color indicates the value of the measure for each configuration.

5.3.4 Randomly perturbed Bravais lattices

To test the sensitivity of each measure to noise present in a lattice, we compute each measure of order for lattices produced by perturbing each point of a perfectly hexagonal lattice. Starting with a perfect hexagonal lattice comprised of 2500 points, each point is displaced from its original position in a random direction (chosen from a uniform distribution) by a distance $\nu\delta$, where ν is a fixed scaling factor and δ is chosen from a uniform distribution. ν ranges from 0.01 to 1. Subsets of the noisy lattice at several ν values are shown in Figure 5.9. The mean and variance of each measure of order over 100 trials, for various values of the scaling factor ν are shown in Figure 5.9 as a function of ν .

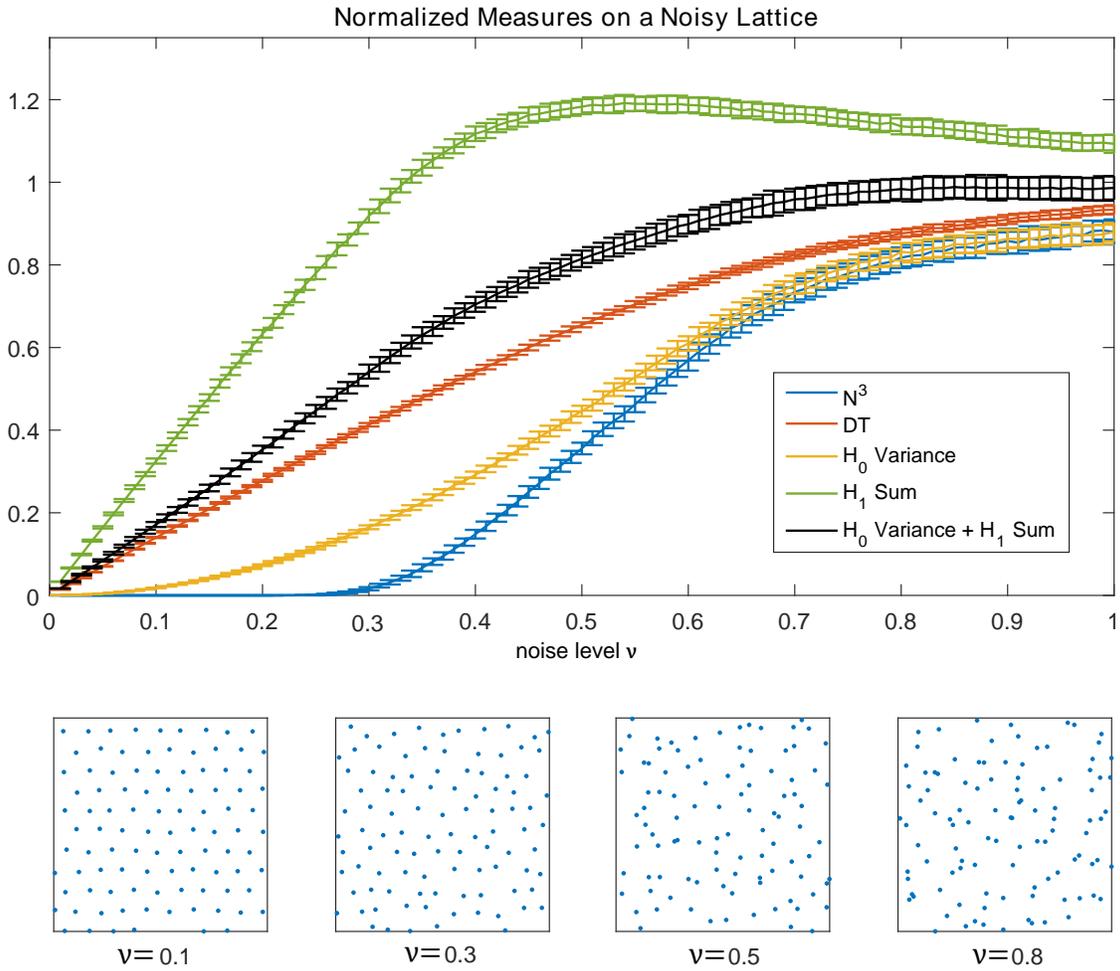


Figure 5.9: Plotted are the normalized measures for perturbed lattices as a function of noise. The error bars indicate the variance of each measure over 100 trials. Each measure has been normalized so that the expected measure has a value of one for points drawn from a uniform distribution rather than perturbed from a perfect lattice. Examples of a section of the lattice at various noise scales are shown below.

In order to fairly compare the various measures, each measure was normalized so that the mean value of the measure is 1 for a set of points chosen randomly with a uniform distribution. The value of the normalization factor was found by computing each measure of order for 500 trials of 2500 points drawn from a uniform distribution on the same square domain used for the randomly perturbed Bravais lattices. The N^3 measure shows little sensitivity for lower levels of noise. There is a large start region in which the center point may be displaced without changing the number of nearest neighbors. In this region, ΣH_1 is more sensitive, which makes ΣH_1 useful for detecting very small perturbations. As the noise increases, the ΣH_1 levels off. However, this is a regime for which $\text{var}(H_0)$ retains sensitivity to changing noise levels. A linear combination of the two PH measures allows us to capitalize on the sensitivity in different regimes. The DT measure displays a similar degree of sensitivity. By $\nu = 0.5$ there is visually very little order, however the N^3 measure and $\text{var}(H_0)$ are much lower than 1, the normalized measure for a noisy sample. This indicates that these measures are picking up some underlying structure of the points.

Classification can be preformed using the normalized $\text{var}(H_0)$ and ΣH_1 scores to create a two dimensional feature vector for each example. We take a subset of the noisy lattice, for ν from 0 to 0.6, incremented by 0.05. $\nu = 0.6$ is a large amount of noise, at this level, lattice structure is not visually distinguishable. With a K-Nearest Neighbors clustering algorithm⁵, a 99.8 % classification accuracy is achieved. The one example that is misclassified in an adjacent class at $\nu = 0.6$. 98.8% accuracy is achieved when the normalized $\text{var}(H_0) + \Sigma H_1$ score is used as a single feature, again with a K-nearest neighbors classifier. If we include all noise levels from $\nu = 0$ to 0.6 incremented by 0.01, classification is 50.4 %, correctly classifying all examples for $\nu \leq 0.1$ and only misclassifying in the next adjacent class for $\nu \leq 0.3$. Using $\text{var}(H_0)$ and ΣH_1 scores separately, a classification accuracy of 74.0 % is achieved, only misclassifying in the next adjacent

⁵The k-nearest neighbor classification algorithm assigns an example a class based on the class of it's k closest neighbors. [67] The metric can vary, here we choose Euclidean distance, and the contribution of each example can be weighted. We consider the 10 nearest neighbors, and weight by the inverse of the squared distance between the example and it's neighbor.

class for $\nu \leq 0.35$, and never misclassifying by more than $\nu = 0.03$. This indicates that these PH statistics are useful in estimating a noise level in a near hexagonal lattice.

5.4 Application to Patterns Produced by Ion Bombardment

In this section, we return to the Bradley-Shipman (BS) equations [105], which models nanoscale pattern formation when a solid surface of a binary compound is bombarded by a broad ion beam. For certain values of the parameters, this system produces hexagonal patterns with defects. This application motivated the development of a persistence based measure for nearly hexagonal patterns. Recall the BS equations govern the behavior of $u(x, y, t)$ and $\phi(x, y, t)$, the deviations of the surface height and surface concentration from their unperturbed, steady-state values and are given by

$$\frac{\partial u}{\partial t} = \phi - \nabla^2 u - \nabla^2 \nabla^2 u + \lambda (\nabla u)^2$$

and

$$\frac{\partial \phi}{\partial t} = -a\phi + b\nabla^2 u + c\nabla^2 \phi + \nu\phi^2 + \eta\phi^3$$

for normal-incidence bombardment. Explicit expressions that relate the dimensionless constants a , b , c , λ , ν and η to the underlying physical parameters may be found in Shipman et al. [105].

Linear stability analysis of the system given by Eqs. (5.4) and (5.4) reveals that the spatially uniform solution $u = 0, \phi = 0$ is stable for values of b above a critical value $b = b_c = (a + c)^2 / (4c)$ [6, 105]. For values of b smaller than b_c , the spatially uniform state is unstable to linear combinations of perturbations of the form

$$\begin{pmatrix} u \\ \phi \end{pmatrix} = \begin{pmatrix} u_* \\ \phi_* \end{pmatrix} \exp(ik \cdot x + \sigma t),$$

where $k \equiv k_x \hat{x} + k_y \hat{y}$, $x \equiv x \hat{x} + y \hat{y}$ and u_* and ϕ_* are constants. $Re(\sigma)$ gives the rate with which the amplitude of the mode grows (for $Re(\sigma) > 0$) or attenuates (for $Re(\sigma) < 0$). For all

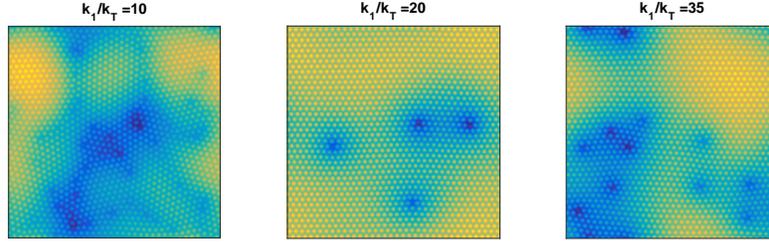


Figure 5.10: Several examples of the resulting surface for $k_1/k_T = 10, 20$ and 35 . There are fewer and less severe defects present at $k_1/k_T = 20$. The orientation of the hexagonal lattice is more consistent across the entire domain as well. $k_1/k_T = 35$ is better ordered than $k_1/k_T = 10$, but there remains some large defects and grain boundaries.

simulations of Eqs. (5.4) and (5.4), the initial conditions are low-amplitude white noise. We employ a Fourier spectral method with periodic boundary conditions and a fourth-order exponential time differencing Runge-Kutta method for the time stepping as the numerical technique [16, 17]. The spatial grid is 256×256 .

5.4.1 Templated Surfaces

It has been shown that beginning with a topographically prepatterned or templated surface can lead to a more highly ordered lattice than would be formed on an initially flat surface [91]. Pearson et. al consider pre patterning with hexagonally ordered arrays of nanoholes, sinusoidal ripples and straight line scratches. For appropriately chosen parameters governing the pre patterning, they note a large improvement in global order. We will investigate sinusoidal templating, using persistence measures to gauge the overall order. Here, we start with simulations with a sinusoidal initial condition. Sinusoidal templating is possible in a laboratory through standard lithographic techniques [91].

$k_T = 2\pi/\lambda_T$ where λ_T is the wavelength with the largest linear growth rate of the surface height. $k_1 = 2\pi/\lambda_1$, where λ_1 is the wavelength of the sinusoid in our template. For a domain that is $[0, L] \times [0, L]$, then $N = \frac{2L}{\lambda_1} \propto k_1$. We restrict to integer N due to periodic boundary conditions. We use a 2D Rips filtration on the locations of the peaks to compute PH. Several examples of surfaces with different choices of sinusoid wavelength are seen in Figure 5.10. We plot the re-

sulting PH scores in Figure 5.11. It can be seen, especially in the $\text{var}(H_0)$, that for $k_1/k_T \in [19, 21]$ there is a significant improvement in the order. This is less clear in ΣH_1 . We notice that there is a distinct increase in the total number of H_1 bars around $k_1/k_T = 18$, and so in the third row, we scale ΣH_1 by \sqrt{N} where N is the number of H_1 bars.

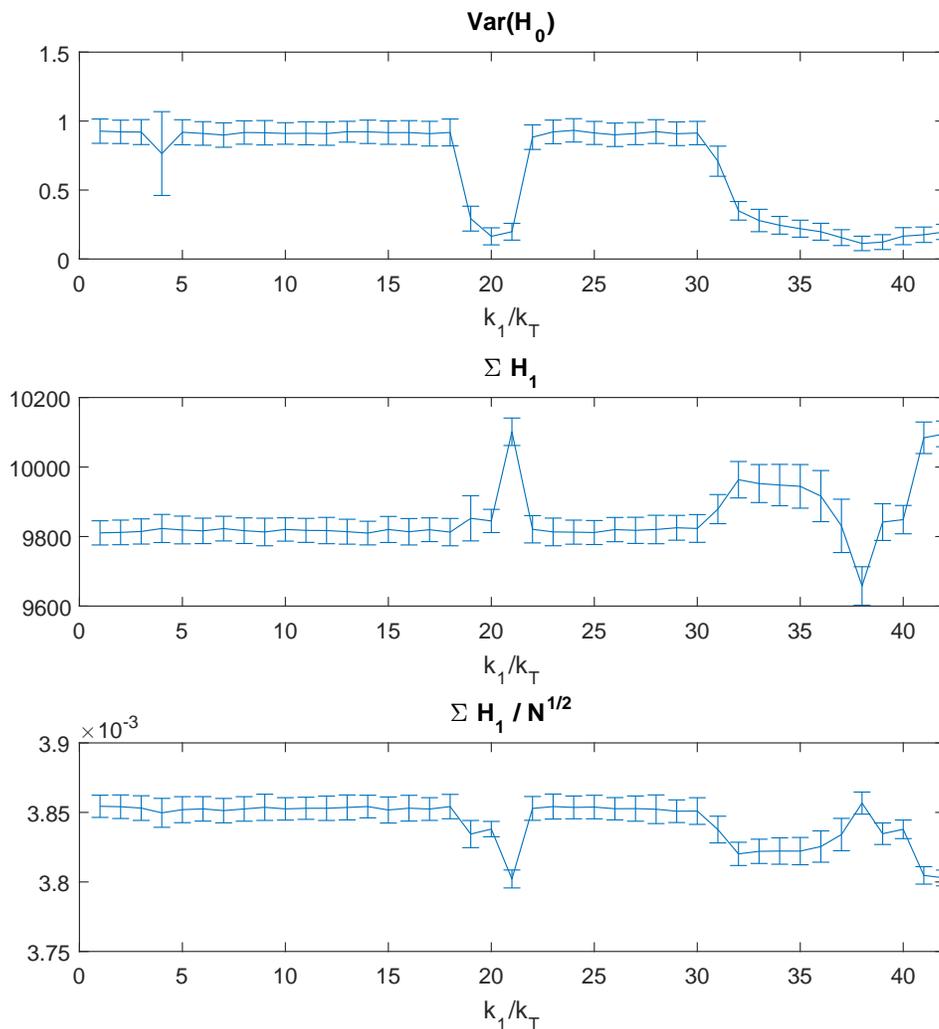


Figure 5.11: PH statistics computed for simulations with sinusoidal templated initial conditions. The final plot scales ΣH_1 by \sqrt{N} , where N is the number of bars. We believe this was a necessary scaling in this case because there was an increase in the number of H_1 bars around $k_1/k_T = 18$.

5.5 Conclusion

Persistent homology is a valuable tool through which one can characterize and quantify underlying topological and geometric order that is present in complex patterns. The ubiquity of data that arises as a result of dynamic processes makes studying the orbits of differential equations and the solutions of PDEs interesting, not only in their own right, but also for model validation, understanding mechanisms that drive patterns and defects of these patterns.

Unique pattern arising in the persistence diagram of a class of one-dimensional discrete dynamical systems—even in chaotic parameter regimes, were connected to the dynamics of the system in Chapter 2. This pattern was shown to be robust to noisy perturbations in both the orbit and the parameter values. We generalized this to the class of unimodal maps in general. Geometric pattern structure, for example of the suncups that form on melting snow fields, can be quantified by using persistence. This gives a multiscale picture of a surface that has underlying large scaled topography that can be difficult to remove from small scale considerations. Sublevel set persistence provides a natural way to compare pyramidal and inverted pyramidal structures that occur when Ge is irradiated with ions. This allows us to clearly see the influence of a model parameter on surface topography. Finally, we exploit the VR filtration to detect perfect hexagonal lattices, developing a PH based measure for quantifying order and comparing this to other common methods of quantifying order.

Persistent homology has allowed us to exploit topological and geometric characteristics of several different complex data sets. We believe there is much more to be learned by using PH as a representation of complex data.

References

- [1] Ailove-grayscott. *GitHub*, 2013.
- [2] H. Adams, T. Emerson, M. Kirby, R. Neville, C. Peterson, P. Shipman, S. Chepushtanova, E. Hanson, F. Motta, and L. Ziegelmeier. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.
- [3] S. Ardanza-Trevijano, I. Zuriguel, R. Arévalo, and D. Maza. Topological analysis of tapped granular media using persistent homology. *Physical Review E*, 89(5):052212, 2014.
- [4] L. S. Block and W. A. Coppel. *Dynamics in one dimension*. Springer, 1992.
- [5] R. Böttger, L. Bischoff, S. Facsko, and B. Schmidt. Quantitative analysis of the order of bion induced dot patterns on ge. *EPL (Europhysics Letters)*, 98(1):16009, 2012.
- [6] R. Bradley and P. Shipman. Spontaneous pattern formation induced by ion bombardment of binary compounds. *Physical Review Letters*, 105, 2010.
- [7] R. M. Bradley and P. D. Shipman. A surface layer of altered composition can play a key role in nanoscale pattern formation induced by ion bombardment. *Applied Surface Science*, 258(9):4161–4170, 2012.
- [8] H. Bruin. Combinatorics of (fibonacci-like) unimodal maps. 1994.
- [9] P. Bubenik. Statistical topological data analysis using persistence landscapes. *The Journal of Machine Learning Research*, 16(1):77–102, 2015.
- [10] J. W. Bush. Quantum mechanics writ large. *Proceedings of the National Academy of Sciences*, 107(41):17455–17456, 2010.
- [11] G. Carlsson. Topology and data. *American Mathematical Society*, 46:255–308, 2009.
- [12] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.

- [13] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Extending Persistence Using Poincaré and Lefschetz Duality. *Foundations of Computational Mathematics*, 9:79–103, 2009.
- [14] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, and Y. Mileyko. Lipschitz functions have L_p -stable persistence. *Foundations of computational mathematics*, 10(2):127–139, 2010.
- [15] P. Collet and J.-P. Eckmann. *Iterated maps on the interval as dynamical systems*. Springer Science & Business Media, 2009.
- [16] S. M. Cox and P. C. Matthews. Exponential time differencing for stiff systems. *Journal of Computational Physics*, 176(2):430–455, 2002.
- [17] R. Craster and R. Sassi. Spectral algorithms for reaction-diffusion equations. *Note del Polo–Ricerca*, (99), 2006.
- [18] M. Cross and H. Greenside. *Pattern formation and dynamics in nonequilibrium systems*. Cambridge University Press, 2009.
- [19] M. C. Cross and P. C. Hohenberg. Pattern formation outside of equilibrium. *Reviews of modern physics*, 65(3):851, 1993.
- [20] R. Cuerno and A.-L. Barabási. Dynamic scaling of ion-sputtered surfaces. *Physical Review Letters*, 74:4746, 1995.
- [21] C. Curto. What can topology tell us about the neural code? *Bulletin of the American Mathematical Society*, 2016.
- [22] S. Day, W. D. Kalies, and T. Wanner. Verified homology computations for nodal domains. *Multiscale Modeling & Simulation*, 7(4):1695–1726, 2009.
- [23] W. De Melo and S. Van Strien. *One-dimensional dynamics*, volume 25. Springer Science & Business Media, 2012.

- [24] F. De Vita, J.-M. Fullana, G. Kirsetter, P.-Y. Lagrée, E. Lane, J. Lopez-Herrera, and S. Popinet. Basilisk, 2017.
- [25] J. S. Deems, S. R. Fassnacht, K. J. Elder, J. S. Deems, S. R. Fassnacht, and K. J. Elder. Fractal distribution of snow depth from lidar data. *Journal of Hydrometeorology*, 7(2):285–297, 4 2006.
- [26] O. Delgado-Friedrichs, V. Robins, and A. Sheppard. Morse theory and persistent homology for topological analysis of 3d images of complex materials. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 4872–4876. IEEE, 2014.
- [27] R. L. Devaney et al. *An introduction to chaotic dynamical systems*. Addison-Wesley Reading, 1989.
- [28] P. Dłotko and T. Wanner. Topological microstructure analysis using persistence landscapes. *Physica D: Nonlinear Phenomena*, 2016.
- [29] J. Eckmann. The mechanism of feigenbaum universality. In *Proceedings of the International Congress of Mathematicians, Berkeley, California, USA*, pages 1263–1267, 1986.
- [30] H. Edelsbrunner and J. Harer. Persistent homology – a survey. *Contemporary Mathematics*, 453:257–282, 2008.
- [31] H. Edelsbrunner and J. Harer. *Computational topology: An introduction*. American Mathematical Society, 2010.
- [32] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*., pages 454–463, 2000.
- [33] H. Edelsbrunner and D. Morozov. Persistent homology: theory and practice. In *Proceedings of the European Congress of Mathematics*, pages 31–50, 2012.

- [34] H. Edelsbrunner and O. Symonova. The Adaptive Topology of a Digital Image. *2012 Ninth International Symposium on Voronoi Diagrams in Science and Engineering*, pages 41–48, 2012.
- [35] S. Facsko, T. Dekorsy, C. Koerdt, C. Trappe, H. Kurz, A. Vogt, and H. L. Hartnagel. Formation of ordered nanoscale semiconductor dots by ion sputtering. *Science*, 285(5433):1551–1553, 1999.
- [36] H. Fang. Studying the lorenz equations with one-dimensional maps from successive local maxima. *Zeitschrift für Physik B Condensed Matter*, 96(4):547–552, 1995.
- [37] S. Fassnacht, I. Oprea, P. Shipman, G. Borleske, F. Motta, and D. Kamin. Geometric methods in the study of snow surface roughness. pages 41–50. *Proceedings of Hydrology Days*, 2015.
- [38] M. J. Feigenbaum. Quantitative universality for a class of nonlinear transformations. *Journal of Statistical Physics*, 19(1):25–52, 1978.
- [39] M. J. Feigenbaum. Universality in complex dynamical systems. *Los Alamos Theoretical Division Annual Report*, pages 98–102, July 1975-September 1976.
- [40] R. Forman. Morse theory for cell complexes. *Advances in mathematics*, 134(1):90–145, 1998.
- [41] R. Forman. A User’s Guide to Discrete Morse Theory. *Séminaire Lotharingien de Combinatoire*, 48, 2002.
- [42] R. Friedrich and S. Gurevich. *Numerische methoden für komplexe systeme*, 2009.
- [43] M. Gameiro, K. Mischaikow, and W. Kalies. Topological characterization of spatial-temporal chaos. *Physical Review E*, 70(3):035203, 2004.
- [44] M. Gameiro, K. Mischaikow, and T. Wanner. Evolution of pattern complexity in the Cahn-Hilliard theory of phase separation. *Acta Materialia*, 53(3):693–704, 2005.

- [45] A. Getling. Structures and dynamics. *Advanced series in nonlinear dynamics*, 11, 1998.
- [46] R. Ghrist. Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.
- [47] R. Ghrist. *Elementary Applied Topology*. Createspace Independent Pub, 2014.
- [48] R. Gilmore. Topological analysis of chaotic dynamical systems. *Reviews of Modern Physics*, 70(4):1455, 1998.
- [49] C. Gollwitzer, I. Rehberg, and R. Richter. Via hexagons to squares in ferrofluids: experiments on hysteretic surface transformations under variation of the normal magnetic field. *Journal of Physics: Condensed Matter*, 18(38):S2643, 2006.
- [50] A. A. Golovin and S. H. Davis. Effect of anisotropy on morphological instability in the freezing of a hypercooled melt. *Physica D: Nonlinear Phenomena*, 116:363–391, 1998.
- [51] R. Groeneveld. *Lactophrys bicaudalis*, 2013.
- [52] J. Harer, R. Bar-On, N. Strawn, C. Tralie, P. Bendich, A. Pieloch, and J. Slaczedek. Tdatools, 2014.
- [53] B. Hashmi, P. D. Shipman, and R. M. Bradley. Highly ordered square arrays of nanoscale pyramids produced by ion bombardment of a crystalline binary material. *Physical Review E*, 93(3):032207, 2016.
- [54] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.
- [55] U. C. Herzfeld, H. Mayer, N. Caine, M. Losleben, and T. Erbrecht. Morphogenesis of typical winter and summer snow surface patterns in a continental alpine environment. *Hydrological Processes*, 2003.
- [56] M. W. Hirsch, S. Smale, and R. L. Devaney. *Differential Equations, Dynamical Systems and An Introduction to Chaos*. Elseiver, second edition, 2009.

- [57] T. K. Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.
- [58] Horn of africa.
- [59] H. M. Jaeger and A. J. Liu. *Far-From Equilibrium Physics: An Overview*, pages 91–110. National Academies Press, 2010.
- [60] T. Kaczynski, K. Mischaikow, and M. Mrozek. *Computational Homology*. Applied Mathematical Sciences. Springer New York, 2004.
- [61] M. Kerber, D. Morozov, and A. Nigmatov. Geometry helps to compare persistence diagrams. *Proceedings of the Workshop on Algorithm Engineering and Experiments*, 2016. Accepted.
- [62] T. Y. Kong and A. Rosenfeld. Digital topology: Introduction and survey, 1989.
- [63] V. A. Kovalevsky. Finite Topology and Image Analysis. *Advances in Electronics and Electron Physics*, 1992.
- [64] M. Kramar, A. Goulet, L. Kondic, and K. Mischaikow. Persistence of force networks in compressed granular media. *Physical Review E*, 87(4):042207, 2013.
- [65] M. Kramár, A. Goulet, L. Kondic, and K. Mischaikow. Quantifying force networks in articulate systems. *Physica D: Nonlinear Phenomena*, 283:37–55, 2014.
- [66] M. Kramár, R. Levanger, J. Tithof, B. Suri, M. Xu, M. Paul, M. F. Schatz, and K. Mischaikow. Analysis of kolmogorov flow and rayleighâ€šbénard convection using persistent homology. *Physica D*, 334:82–98, 2016.
- [67] M. Kubat. *An Introduction to Machine Learning*. Springer International Publishing, 2015.
- [68] H. Kurtuldu, K. Mischaikow, and M. F. Schatz. Measuring the departures from the Boussinesq approximation in Rayleighâ€šBénard convection experiments. *Journal of Fluid Mechanics*, 682:543–557, 2011.

- [69] C. Letellier and R. Gilmore. *Topology and Dynamics of Chaos: In Celebration of Robert Gilmore's 70th Birthday*. World Scientific Publishing Company, 2013.
- [70] L. Levitov. Long-range order, symmetry, and soft modes. 2003. Lecture.
- [71] T.-Y. Li and J. A. Yorke. Period three implies chaos. *The American Mathematical Monthly*, 82(10):985–992, 1975.
- [72] E. N. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963.
- [73] S. Mátéfi-Tempfli, M. Mátéfi-Tempfli, and L. Piraux. Characterization of nanopores ordering in anodic alumina. *Journal of Thin Solid Films*, 516(12):3735–3740, 2007.
- [74] R. M. May. Simple mathematical models with very complicated dynamics. *Nature*, 261(5560):459–467, 1976.
- [75] J. D. Meiss. *Discrete Dynamical Systems*. in preparation.
- [76] N. Metropolis, M. Stein, and P. Stein. On finite limit sets for transformations on the unit interval. *Journal of Combinatorial Theory, Series A*, 15(1):25–44, 1973.
- [77] J. Milnor. *Morse Theory*, volume 51. Princeton University Press, 2016.
- [78] K. Mischaikow. Conley index theory. In *Dynamical systems*, pages 119–207. Springer, 1995.
- [79] K. Mischaikow and V. Nanda. Morse theory for filtrations and efficient computation of persistent homology. *Discrete & Computational Geometry*, 50(2):330–353, 2013.
- [80] K. A. Mitchell and T. Tiedje. Growth and fluctuations of suncups on alpine snowpacks. *Journal of Geophysical Research*, 115(F4):F04039, 12 2010.
- [81] F. C. Motta. *Discrete-time topological dynamics, complex Hadamard matrices, and oblique-incidence ion bombardment*. PhD thesis, 2014.

- [82] F. C. Motta, R. Neville, P. D. Shipman, D. Pearson, and R. M. Bradley. Topological measures of order for nearly hexagonal lattices. *preprint*, 2017.
- [83] F. C. Motta, P. D. Shipman, and R. M. Bradley. Highly ordered nanoscale surface ripples produced by ion bombardment of binary compounds. *Journal of Physics D: Applied Physics*, 45(12):122001, 2012.
- [84] M. Mrozek and K. Wójcik. Discrete version of a geometric method for detecting chaotic dynamics. *Topology and its Applications*, 152(1-2):70–82, 2005.
- [85] T. Nakamura, Y. Hiraoka, A. Hirata, E. G. Escobar, and Y. Nishiura. Persistent homology and many-body atomic structure for medium-range order in the glass. *Nanotechnology*, 26(30):304001, 2015.
- [86] M. Natiello and H. Solari. *The User's Approach to Topological Methods in 3d Dynamical Systems*. World Scientific, 2007.
- [87] S. Obukhov. Self-organized criticality: Goldstone modes and their interactions. *Physical review letters*, 65(12):1395, 1990.
- [88] E. Ott. *Chaos in Dynamical Systems*. Cambridge University Press, second edition, 2002.
- [89] X. Ou, A. Keller, M. Helm, J. Fassbender, and S. Facsko. Reverse epitaxy of ge: Ordered and faceted surface patterns. *Physical Review Letters*, 111(1), 2013.
- [90] D. Papageorgiou and Y. Smyrlis. The route to chaos for the kuramoto-sivashinsky equation. *Theoretical and Computational Fluid Dynamics*, 3(1):15–42, 1991.
- [91] D. A. Pearson, R. M. Bradley, F. C. Motta, and P. D. Shipman. Producing nanodot arrays with improved hexagonal order by patterning surfaces before ion sputtering. *Physical Review E*, 92(6):062401, 2015.

- [92] J. A. Perea and J. Harer. Sliding windows and persistence: An application of topological methods to signal analysis. *Foundations of Computational Mathematics*, pages 1–40, 2013.
- [93] S. Pichler, M. Bodnarchuk, M. Kovalenko, M. Yarema, G. Springholz, D. Talapin, and W. Heiss. Evaluation of ordering in single-component and binary nanocrystal superlattices by analysis of their autocorrelation functions. *ACS Nano*, 5(3):1703–1712, 2011.
- [94] M. Pourfard, K. Faez, and S. H. Tabaian. Autocorrelation-based method for characterization of the self-hexagonal lattice. *Journal of Physical Chemistry C*, (117):17225–17236, 2013.
- [95] G. Radons, W. Just, and P. HÄd’ussler. *Collective dynamics of nonlinear and disordered systems*. Springer, 2005.
- [96] J. Reininghaus, D. Günther, I. Hotz, S. Prohaska, and H.-C. Hege. Tadd: A computational framework for data analysis using discrete morse theory. In *International Congress on Mathematical Software*, pages 198–208. Springer, 2010.
- [97] J. Reininghaus, S. Huber, U. Bauer, and R. Kwitt. A stable multi-scale kernel for topological machine learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4741–4748, 2015.
- [98] J. Reininghaus, N. Kotava, D. Guenther, J. Kasten, H. Hagen, and I. Hotz. A scale space based persistence measure for critical points in 2d scalar fields. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2045–2052, 2011.
- [99] R. Gilmore and M. Lefranc. *The Topology of Chaos: Alice in Stretch and Squeezeland*. Wiley, 2012.
- [100] V. Robins, P. J. Wood, and A. P. Sheppard. Theory and algorithms for constructing discrete morse complexes from grayscale digital images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 2011.

- [101] M. Rost and J. Krug. Anisotropic kuramoto-sivashinsky equation for surface growth and erosion. *Physical Review Letters*, 75:3894, 1995.
- [102] E. Sander and J. A. Yorke. Connecting period-doubling cascades to chaos. *International Journal of Bifurcation and Chaos*, 22(02):1250022, 2012.
- [103] S. Scherer and M. R. Schindler. Spontaneous symmetry breaking and the goldstone theorem. In *A Primer for Chiral Perturbation Theory*, pages 49–64. Springer, 2011.
- [104] V. B. Shenoy, W. L. Chan, and E. Chason. Compositionally modulated ripples induced by sputtering of alloy surfaces. *Phys. Rev. Lett.*, 98:256101, Jun 2007.
- [105] P. D. Shipman and R. M. Bradley. Theory of nanoscale pattern formation induced by normal-incidence ion bombardment of binary compounds. *Physical Review B*, 84(8):085420, 2011.
- [106] G. I. Sivashinsky. Instabilities, pattern formation, and turbulence in flames. *Annual Review of Fluid Mechanics*, 15(1):179–199, 1983.
- [107] Y. S. Smyrlis and D. T. Papageorgiou. Predicting chaos for infinite dimensional dynamical systems: The kuramoto-sivashinsky equation, a case study. *Proceedings of the National Academy of Sciences*, 88(24):11129–11132, 1991.
- [108] S. H. Strogatz. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering*. Addison-Wesley Publishing Company, 1994.
- [109] H. L. Swinney. Emergence and evolution of patterns. In *AIP Conference Proceedings*, pages 3–22. IOP Institute of Physics Publishing LTD, 1999.
- [110] A. Tausz, M. Vejdemo-Johansson, and H. Adams. JavaPlex: A research software package for persistent (co)homology. In H. Hong and C. Yap, editors, *Proceedings of ICMS 2014*, Lecture Notes in Computer Science 8592, pages 129–136, 2014. Software available at <http://appliedtopology.github.io/javaplex/>.

- [111] H. Thunberg. Periodicity versus chaos in one-dimensional dynamics. *SIAM review*, 43(1):3–30, 2001.
- [112] T. Tiedje, K. A. Mitchell, B. Lau, A. Ballestad, and E. Nodwell. Radiation transport model for ablation hollows on snowfields. *Journal of Geophysical Research: Earth Surface*, 111(F2), 2006.
- [113] C. Topaz, L. Ziegelmeier, and T. Halverson. Topological data analysis of biological aggregation models. *PloS One*, 10(5):e0126383, 2015.
- [114] M. Vejdemo-Johansson. Sketches of a platypus: Persistent homology and its algebraic foundations. *ArXiv e-prints*, 2012.
- [115] S. K. Verovšek. Tropical coordinates on the space of persistence barcodes. *arXiv preprint arXiv:1604.00113*, 2016.
- [116] J. Villain. Continuum models of crystal growth from atomic beams with and without desorption. *J. Phys. I France*, 1:19–42, 1991.
- [117] H. Wagner, C. Chen, and E. Vuçini. Efficient computation of persistent homology for cubical data. In *Topological Methods in Data Analysis and Visualization II*, pages 91–106. Springer, 2012.
- [118] T. Wanner. Topological analysis of the diblock copolymer equation. In *Mathematical Challenges in a New Phase of Materials Science*, pages 27–51. Springer, 2016.
- [119] J. Wei and M. Winter. *Mathematical aspects of pattern formation in biological systems*, volume 189. Springer Science & Business Media, 2013.
- [120] D. Wu, L. Duan, and Q. Kang. Wavenumber selection by Bénard-Marangoni convection at high supercritical number. *Chinese Physics Letters*, 34(5):054702, 2017.
- [121] Y.-N. Young and H. Riecke. Penta-hepta defect chaos in a model for rotating hexagonal convection. *Physical review letters*, 90(13):134502, 2003.