

Persistent Homology to Characterize Dynamics - a Statistical Analysis

Master's Thesis submitted in partial fulfilment of the requirements for
the award of the degree of Master's of Science in Physics.

by

Joel Prakash Stephen
MSc/07/16

Under the Joint Supervision of:

Dr. Prince Ganai
National Institute of Technology, Srinagar
&
Dr. N. Nirmal Thyagu
Vellore Institute of Technology, Chennai Campus

Submitted to:



Department of Physics
National Institute of Technology
Hazratbal, Srinagar, J & K - 190006

CERTIFICATE

This is to certify that the thesis entitled "**Persistent Homology to Characterize Dynamics - a Statistical Analysis**" submitted by **Joel Prakash Stephen** (Enroll. No: **MSc- 07/16**) to the National Institute of Technology Srinagar, Jammu and Kashmir for the award of the degree of M.Sc. Physics is a bonafide record of research work carried out by him under my supervision. The contents of the thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Place: Srinagar
Date: 29th June 2018

Dr. Prince Ganai
Department of Physics
National Institute of Technology Srinagar
Jammu & Kashmir, India - 190006

DECLARATION

I hereby declare that this Master's Thesis titled 'Persistent Homology to Characterize Dynamics - A Statistical Approach', submitted to the Department of

I hereby declare that the thesis entitled "**Persistent Homology to Characterize Dynamics - A Statistical Analysis**" is an authentic record of research work carried out by me, under the joint supervision of **Dr. Prince Ganai**, Department of Physics, NIT Srinagar, Hazratbal, Jammu and Kashmir, India and **Dr. N. Nirmal Thyagu**, Division of Physics, School of Advanced Sciences, VIT Chennai, India. The matter embodied in this thesis was not been submitted for the award of any other degree. Keeping in view, the general practice of reporting scientific observations, due citation has been made wherever obligated. Any omission that might have occurred due to oversight or error in judgment is deeply regretted.

Place: Srinagar

Date: 29th June 2018

Joel Prakash Stephen

Enrollment No. - **MSc/07/16**

To all those who took interest in me and played a role in building me up.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my project guide Dr. N. Nirmal Thyagu at Vellore Institute of Technology - Chennai Campus for his patience, motivation and mentor-ship throughout the course of the project. His calm demeanour and reassurance helped me on the hard days and I shall be always grateful to him for investing in me and for giving me a good experience of what research really is. His advice and guidance is what gave direction to this piece of work and shaped this project to be what it is.

My sincere thanks go to my teacher and guide Dr. Prince Ganai who throughout my Master's course with his infectious enthusiasm, encouraged me to admire the beauty of mathematics within the regimes of physics. I am thankful for his insight and guidance throughout the duration of my course and during this project.

Besides my professors, I would like to thank my friends and family who supported me in innumerable ways during my stay at Chennai and made my time one worth cherishing. I'd also like to thank all my batch-mates who encouraged me during this time.

My parents' support and encouragement have played an integral role during the course of this project. I am very grateful to them.

Last but not the least, I'd like to thank God for enabling me to complete this work.

ABSTRACT

Persistent Homology is a topological data analysis tool that attempts to explain the global shape of a point cloud dataset by primarily studying the local features of the point cloud dataset. This report analyzes and interprets a multi-dimensional nonlinear dynamical system, namely the duffing oscillator using persistent homology. The topological structure of the phase portraits are studied and correlated with the period of the oscillator. The lifetimes obtained from the filtration are then fed into ROC (Receiver Operator Characteristic), a statistical analysis tool which allows one to come up with a threshold to differentiate between noise and data.

CONTENTS

CHAPTER 1 – OVERVIEW	1
1.1 Topological Data Analysis (TDA)	1
1.2 Where do TDA and Dynamics meet?	2
1.3 Statistical Analysis	2
CHAPTER 2 – HOMOLOGY	3
2.1 Simplex	3
2.2 Simplicial Complex	4
2.3 Chains, Cycles and Boundaries	4
2.4 Betti Numbers	12
2.5 Vietoris-Rips Complex	12
2.6 Filtration and Persistence	13
2.7 Application of Persistent Homology	14
2.8 A Visualization of Noise and Data	17
2A Appendix - Homology	19
CHAPTER 3 – NONLINEAR DYNAMICS	22
3.1 Introduction	22
3.2 Duffing Oscillator	23
3.3 Phase-space Plots	24
3.4 Periodicity	28
CHAPTER 4 – APPLICATION OF PERSISTENT HOMOLOGY	29
4.1 Introduction	29
4.2 Generating the point cloud dataset	29

4.3	Input to Javaplex	30
4.4	Output of Javaplex	32
4.5	Barcode Plots	33
4.6	Birth Time, Death Time and Lifetime Data	39
CHAPTER 5 – STATISTICAL ANALYSIS		51
5.1	Motivation	51
5.2	Receiver Operating Characteristic	51
5.3	Application of ROC	55
5.4	ROC Program	57
5.5	Choosing Cut-offs	59
CHAPTER 6 – CONCLUSION		61
Appendices		63
CHAPTER A – PROGRAMS		64
REFERENCES		71

CHAPTER 1

Overview

"As our circle of knowledge expands, so does the circumference of darkness surrounding it."

- Albert Einstein

With the exponential growth of data all around the world, mathematicians and statisticians have been inventing novel methods to analyze and interpret data. This project revolves around bringing one of these data analysis methods, namely persistent homology into the regime of analyzing and interpreting dynamics.

1.1 Topological Data Analysis (TDA)

Topological data analysis has been making its way into the physics world for a while now. These powerful tools which have been proven to be extremely useful in the analysis of data generated in various fields like computer networks, neural networks and social networks.

Topology is the field of mathematics that studies of the shapes of spaces by quantifying shape. Topological data analysis attempts to simplify a large point cloud dataset and reduce it to the bare-bones without losing the underlying topological structure of the data. This method attempts to predict the global features of the space by studying the local features of the space. [1] Such an approach to studying data, though in it's nascent stages especially in the case of interpreting dynamics, may prove to be extremely

insightful as new methods are introduced. Two popular methods of topological data analysis are Mapper[2] and Persistent Homology. This report shall mainly focus on the latter.

Persistent Homology

Persistent Homology is employed by researchers across disciplines to find the underlying "shape" of a point cloud dataset. This method involves "meshing" the entire space of the data with simplicial complexes.

1.2 Where do TDA and Dynamics meet?

Lately Topological Data analysis has made its way into the regime of Dynamics.[3] [4] and have attempted to study the topological structure of the data extracted from the phase plots of various dynamical systems. Such an analysis of the topology of data extracted from a phase plot can be useful to predict the period of oscillation of a system and will also be useful to study bifurcations. This document addresses the former.[3]

1.3 Statistical Analysis

The data produced by TDA algorithms produce both data and noise. Thus, we turn to statistical analysis for defining a threshold to differentiate between noise and data in the system. Since the data requires a binary classification, ROC (Receiver Operating Characteristic) is a good option to opt for.

CHAPTER 2

Homology

2.1 Simplex

The q -simplex σ is defined as the convex hull of $q+1$ independent points in a q -dimensional subspace of \mathbb{R}^m . The points are called “vertices” and these are joined by “edges”.

Geometrically, a q -simplex can be visualized as the complete graph of $q + 1$ vertices which is solid in q -dimensions. Topologically, the n -simplex is equivalent to D^n , the n -dimensional ball. In general, a q -simplex is formed by $q + 1$ number of $q - 1$ dimensional vertices which are joined together to form a q -dimensional structure.

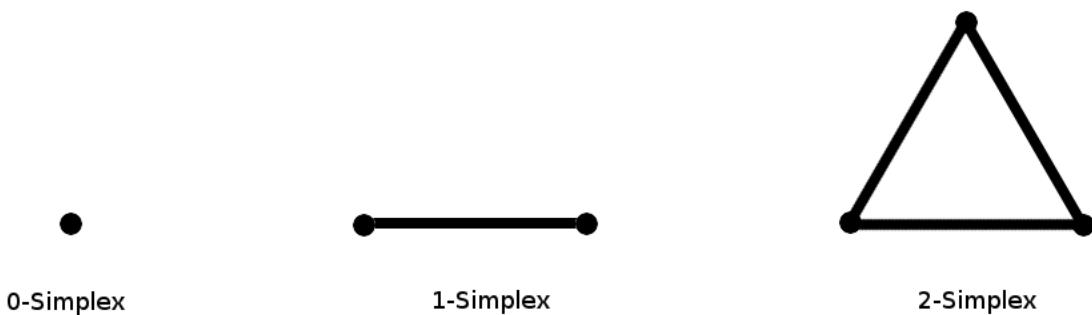


Figure 2.1: 0,1 and 2 simplices

2.2 Simplicial Complex

A simplicial complex \mathbf{K} in \mathbb{R}^m is a collection of simplices in \mathbb{R}^m such that

- * Every every face of a simplex \mathbf{K} is in \mathbf{K} .
- * The intersection of two or more simplices of \mathbf{K} is a face of each of them.

Informally, A simplicial complex is formed by connecting simplices such that they are connected through a common simplex (vertex, edge, face, etc).

2.3 Chains, Cycles and Boundaries

In any simplicial complex, it is possible to construct paths from one "node" to the other through "edges". Such paths are called chains. These chains can have different dimensions. A path consisting of 1-simplices is a 1-dimensional chain. Similarly, a path consisting of 2-simplices will form a 2-dimensional chain. Thus we can have n-dimensional chains in a simplicial complex that contains n-simplices.

There are two simple classifications of chains that are useful to define homology groups. There are chains that form "cycles" and there are those that don't. A n-cycle is a n-chain with empty boundary, thus $\partial(C) = 0$ for a cycle. Since ∂ commutes with addition, we have a group of n-cycles, denoted as $Z_n = Z_n(K)$, which is a subgroup of the group of n-chains. In other words, the group of n-cycles is the kernel of the n-th boundary homomorphism, $Z_n = \ker(\partial_n)$. Since cycle groups are a subgroup of chain groups, they are abelian. Consider $n = 0$ as an example. The boundary of every vertex is zero ($C_1 = 0$), hence, $Z_0 = \ker(\partial_0) = C_0$, for $n > 0$, but, Z_n is usually not all of C_n . An n-boundary is a n-chain that is the boundary of an $(n+1)$ -chain, $c = \partial(d)$ with $d \in C_{n+1}$. Since ∂ commutes with addition, we have a group of n-boundaries, denoted by $B_n = B_n(K)$, which is again a subgroup of all n-chains. In other words, the group of n-boundaries is the image of the $(n+1) - th$ boundary homomorphism, $B_n = \text{im}(\partial_{n+1})$. Since the chain groups are abelian and since boundaries are subgroups of chain groups they are also

definitely abelian. Consider, for example the case where $n = 0$. Every 1-chain consists of some number of edges, each with two endpoints. Taking the boundary cancels duplicate endpoints in pairs, leaving an even number of distinct vertices. A fundamental property of homology is that the boundary of a boundary is always zero. [5]

We begin with a directed graph with nodes and edges as shown in the Figure. There are three closed loops that are visible, $a+b+c$, $a+b+d$ and $c-d$. All $1 - \text{dimensional}$ chains are represented by the Free Abelian group on (directed) a,b,c and d. Thus any linear combination of a,b,c and d will correspond to a $1 - \text{dimensional}$ chain in the graph. We name these $1 - \text{dimensional}$ chains as \mathbf{C}_0 .[6]

Let us assume a graph G such that the nodes and edges are as shown in the figure.

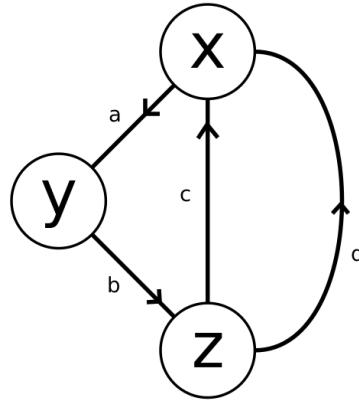


Figure 2.2: The Space - X_1

$$\partial(a) = y - x$$

$$\partial(b) = z - y$$

$$\partial(c) = x - z$$

$$\partial(d) = x - z$$

C_0 : Integral combinations of 'x', 'y' and 'z'. These correspond to $0 - \text{dim}$ chains in the graph. Example - $x + y + z$, $4x - 5y + 2z$, $z - y - x$.

C_1 : Integral combinations of 'a', 'b', 'c' and 'd'. These correspond to $1 - \text{dim}$ chains in the graph. Example - $a + b + c$, $2c - d + a$, $c - d$. (*Note- These are not necessarily closed chains.*)

$$\text{A path becomes a closed chain/cycle} \iff \partial(\text{path}) = 0 .$$

The above statement can be easily verified as shown here using an example. Let us take the paths $\mathbf{a} + \mathbf{b} + \mathbf{c}$ and $\mathbf{a} + \mathbf{c}$.

Applying the ' ∂' operator on $\mathbf{a} + \mathbf{b} + \mathbf{c}$ we get :

$$\partial(a + b + c) = y - x + z - y + x - z = 0$$

As we can see from the above example, all the nodes cancel out and we are left with zero when we apply the ' ∂' operator on a closed loop.

Now let us apply the ' ∂' operator on $\mathbf{a} + \mathbf{c}$:

$$\partial(a + c) = y - x + x - z = 0$$

We can see that in the absence of a closed loop, the ' ∂' operator does not send the path to zero.

So, to find all the cycles/closed loops present in a graph, we can say that :

$$\partial(\alpha a + \beta b + \gamma c + \delta d) = 0 \quad (2.1)$$

$$\alpha\partial(a) + \beta\partial(b) + \gamma\partial(c) + \delta\partial(d) = 0 \quad (2.2)$$

$$\alpha(y - x) + \beta(z - y) + \gamma(x - z) + \delta(x - z) = 0 \quad (2.3)$$

$$x(\gamma + \delta - \alpha) + y(\alpha - \beta) + z(\beta - \gamma - \delta) = 0 \quad (2.4)$$

So,

$$-\alpha + \gamma + \delta = 0 \quad (2.5)$$

$$\alpha - \beta = 0 \quad (2.6)$$

$$\beta - \gamma - \delta = 0 \quad (2.7)$$

Thus the augmented matrix corresponding to the system of equations is:

$$\begin{pmatrix} -1 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & -1 \end{pmatrix} \quad (2.8)$$

Now, applying Similarity Transformations, we can achieve :

$$\begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (2.9)$$

We can now say that:

$$A\vec{X} = \vec{0} \quad (2.10)$$

$$\begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.11)$$

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix} = r \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} + s \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} \quad (2.12)$$

where we have treated r and s as parameters. Such that:

$$\gamma = r \quad (2.13)$$

$$\delta = s \quad (2.14)$$

Hence we can see that linear combinations of these two basis vectors span the null space corresponding to matrix A .

$$\begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \text{ corresponds to } a + b + c \text{ and } \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} \text{ corresponds to } a + b + d.$$

Thus it can be concluded that all cycles in the graph can be described by linear integral combinations of $a + b + c$ and $a + b + d$. Thus for this particular graph we can say that these are the Homology Group Generators.

Mathematically,

$$\ker(\partial) = \langle a + b + c, a + b + d \rangle \quad (2.15)$$

Group of cycles $\simeq \mathbb{Z} \oplus \mathbb{Z}$

\mathbb{H}_1 = Space of 1-dim Homology Groups.

Note: The first impression would be to assume that there are three closed loops,yet

it is interesting to notice that the cycle $c - d$ is not a basis vector. This is because $(a + b + c) - (a + b + d) = c - d$. Thus as proved above, all cycles can be described as linear combinations of $a + b + c$ and $a + b + d$.

As already established,

C_0 : Integral combinations of 'x', 'y' and 'z'. These correspond to $0 - \dim$ chains in the graph. Example - $x + y + z$, $4x - 5y + 2z$, $z - y - x$.

C_1 : Integral combinations of 'a', 'b', 'c' and 'd'. These correspond to $1 - \dim$ chains in the graph. Example - $a + b + c$, $2c - d + a$, $c - d$. (*Note- These are not necessarily closed chains.*)

If the space is modified (\mathbb{X}_2) by adding a 2-cell between the edges $c - d$ as shown in *Figure 2.3* and the orientation is set as clockwise. The orientation is set such that:

$$\partial(A) = c - d \quad (2.16)$$

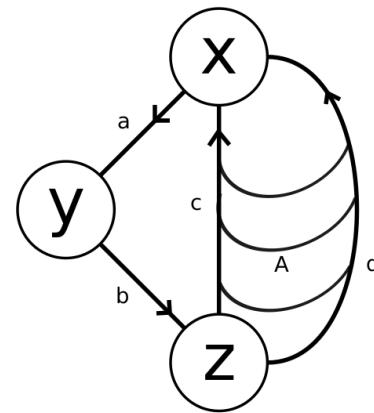


Figure 2.3: The Space - X_2 , after a 2-cell is added between c and d .

C_0 : Integral combinations of A . These correspond to $2 - \dim$ chains in the graph. Example - A , $-4A$, $20A$.

After the addition of A , $c - d$ is no longer a valid cycle because it does not enclose a hole anymore and thus it is homotopically reducible to a point. This alters the Homology of the space.

Since,

$$\partial(A) = c - d \quad (2.17)$$

$$\partial(A) = 0 \quad (2.18)$$

$$\Rightarrow c = d \quad (2.19)$$

Thus since $c = d$ it can be concluded that $a + b + c = a + b + d$. Thus it can be easily observed that it is meaningless to conclude that the basis states for all cycles are $a + b + c$ and $a + b + d$. Since Homotopically, A can be shrunk to a single point and due to the reasons shown above it can be concluded that $a + b + c$ and $a + b + d$ are essentially the same path.

The cycles modulo the boundaries give the homology of the space. Therefore the Homology = $\mathbb{Z}_2 / \mathbb{Z} \simeq \mathbb{Z}$.

Now, in space \mathbb{X}_3 Adding another 2-cell, ' B ' as shown in *Figure 2.4* to the space such that the boundary of B is also $c - d$. This can be visualized by imagining one 2-cell above and one 2-cell below the plane.

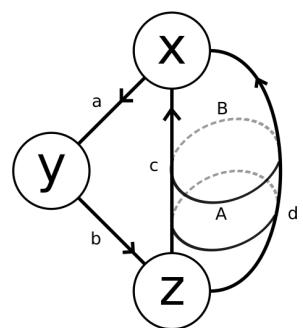


Figure 2.4: The Space - \mathbb{X}_3

Now,

$$\partial(B) = 0 \quad (2.20)$$

$$C_2 = \langle A, B \rangle \quad (2.21)$$

$$\partial_2(\mathbf{C}_2) \rightarrow \mathbf{C}_1 \quad (2.22)$$

$$\partial_1(\mathbf{C}_1) \rightarrow \mathbf{C}_0 \quad (2.23)$$

$$\partial(A) = c - d \quad (2.24)$$

$$\partial(B) = c - d \quad (2.25)$$

$$\Rightarrow \partial_2(A) = \partial_2(B) \quad (2.26)$$

$$\partial_2(A) - \partial_2(B) = 0 \quad (2.27)$$

$$\partial(A - B) = 0 \quad (2.28)$$

Thus we can conclude that the second homology, $\mathbb{H}_2(\mathbb{X}_3)$ is generated by $A - B$.

$$\mathbb{H}_1 = \mathbb{Z}_1 / \mathbb{B}_1 \quad (2.29)$$

$$\mathbb{H}_1 = \frac{\ker(\partial_1)}{\text{im}(\partial)_2} \quad (2.30)$$

The image of ∂_2 remains the same, thus

$$\mathbb{H}_1 = \frac{\mathbb{Z} + \mathbb{Z}}{\mathbb{Z}} \simeq \mathbb{Z} \quad (2.31)$$

The $n - th$ Homology group is the $n - th$ cycle group modulo the $n - th$ boundary group, $\mathbf{H}_n = \mathbf{Z}_n / \mathbf{B}_n$. The $n - th$ Betti Number is the rank of this group, $\beta_n = \text{rank } \mathbf{H}_n$. Also, Betti Numbers correspond to homology group generators.

2.4 Betti Numbers

The n -th Betti number is the rank of the n -th Homology group. Geometrically Betti Numbers can be explained this way - the n -th Betti number corresponds to the number of $n+1$ dimensional voids in the structure. $\beta_n = H_n$.

As shown in *Figure 2.5* the topological features of any structure enclosing a single loop are all equivalent.

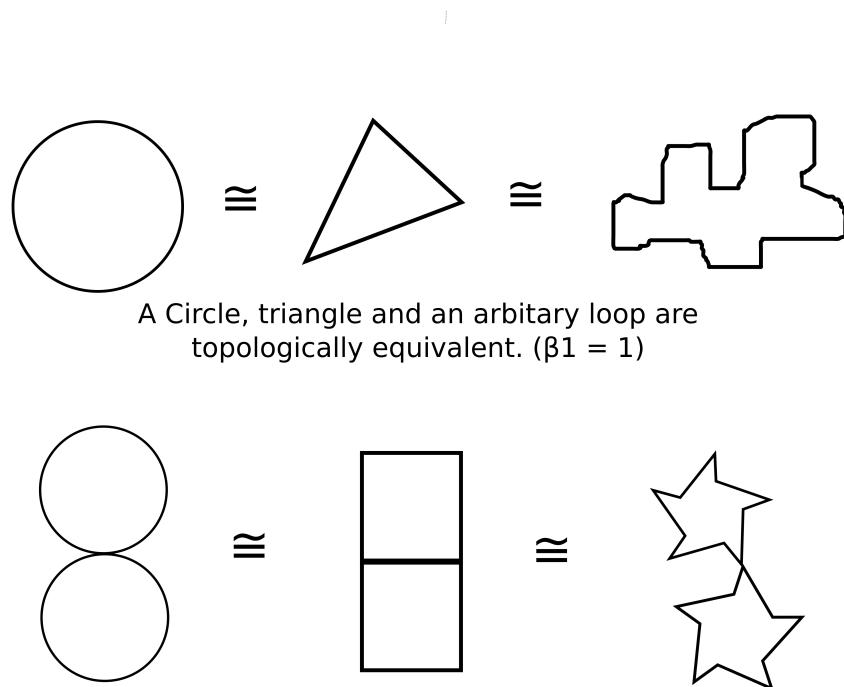


Figure 2.5: β_1 of some structures exhibiting topological equivalence.

2.5 Vietoris-Rips Complex

The Vietoris-Rips (VR) complex is a simplicial complex constructed upon a point cloud data set. Let D be a discrete set of points paired with a metric d . Then the Vietoris-Rips complex for D and parameter ϵ is defined as $VR(D, \epsilon)$, to be the simplicial complex whose vertex set is D and where x_1, x_2, \dots, x_k denotes a k – simplex if and only if $d(x_i, x_j) \leq \epsilon$

for all $0 \leq i, j \leq k$. [1] The entire data set becomes the vertex set, while the set of edges are controlled by the distance parameter ϵ .

The method of construction follows the steps given below-

1. Assume all points in the point cloud dataset to be vertices.
2. Draw spheres of radius ϵ around each point in the point cloud data set.
3. When two spheres intersect, introduce an edge between those two corresponding vertices.

In this way, the simplicial complex is formed. This simplicial complex is called the Vietoris-Rips complex.

2.6 Filtration and Persistence

Filtration is an increasing sequence of nested simplicial complexes. From the definition of the VR complex it is evident that if $\epsilon < \epsilon'$ then $VR(D, \epsilon) \subset VR(D, \epsilon')$. Filtration is based upon that fact.

A filtered simplicial complex is the set $VR(D, \epsilon_i)$ where $\epsilon_i = \epsilon_1, \epsilon_2, \dots, \epsilon_n$ such that $\epsilon_1 < \epsilon_2 < \dots < \epsilon_n$. Simultaneously considering a large number of simplicial complexes with a wide range of ϵ values so that we can get the homological information for a large range of ϵ values. When the homology group generators of a "filtration" are plotted on a "Barcode" plot, it brings out the concept of "birth" and "death" of the homology group generators. In the "dimension 1" section of the barcode plot in *Figure 2.6* it can be observed that there are a large number of horizontal lines, and each line stands for a homology group generator that comes into existence at some value of ϵ and then ceases to exist at some value of ϵ . These points are known as the "birth times" and the "death times" of the homology group generators. The term "Lifetime" is thus quite intuitively defined as the difference between the death time and the birth time.

The idea of persistence revolves around the fact that the shorter lifetimes correspond to noise and the longer lifetimes correspond to data. Thus, we can say that if a homology group generator persists for a long range of ϵ values through the filtration it can be counted as a valid "betti number", hence, the name 'persistent' homology.

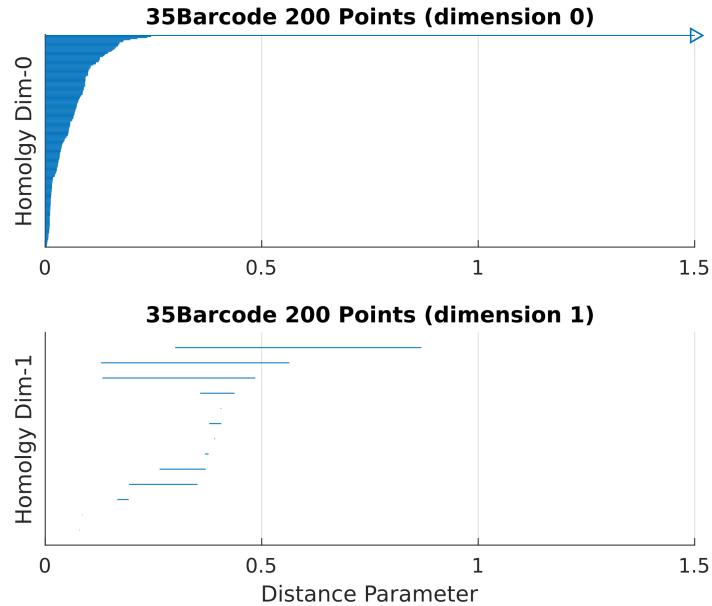


Figure 2.6: An example of a barcode plot.

2.7 Application of Persistent Homology

Provided a point cloud dataset, it is possible to study the underlying topology of the dataset by employing Persistent Homology. The steps are simple, each point is considered a vertex and then a filtration is obtained by constructing VR complexes.

Using such a scheme, it has been demonstrated, how one can find the betti numbers using persistent homology. An example of points sampled on a torus has been shown here in [Figure 2.7](#). Creating VR complexes and obtaining a barcode plot after filtration shows that it is possible to accurately predict the betti numbers of the data that has been sampled. As shown in [Figure 2.7](#) it can be seen how the evidently long lifetimes

correspond to betti numbers of the sampled data.

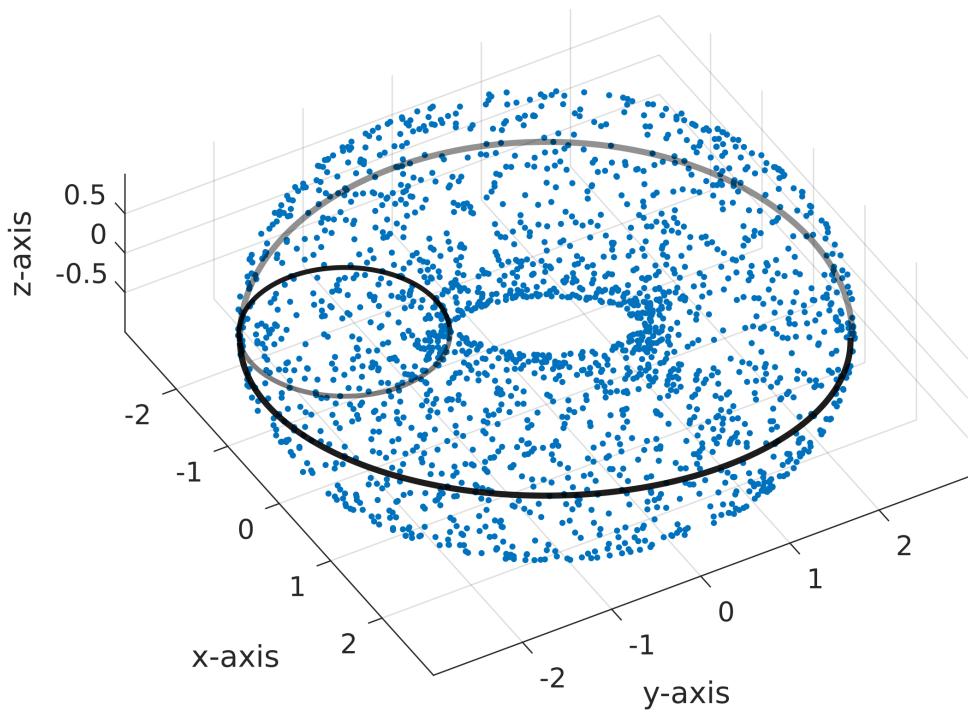


Figure 2.7: 1800 Points sampled from a torus.

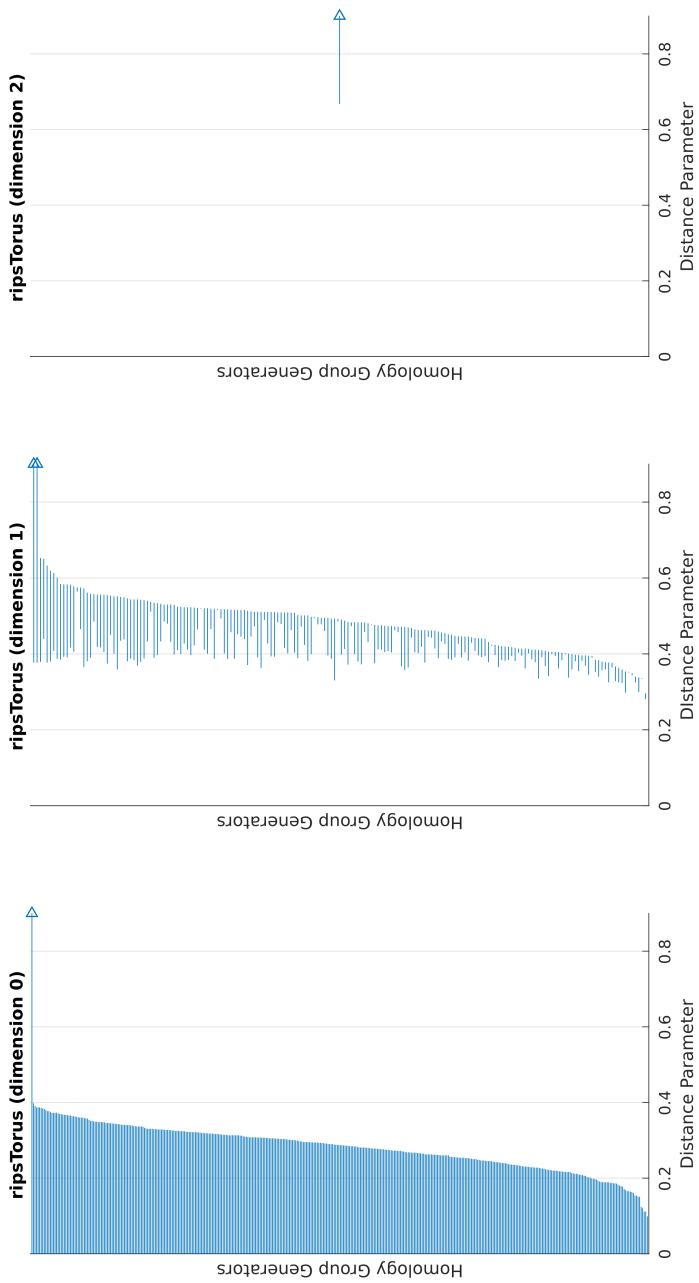


Figure 2.8: Barcode Plot for 1800 data points sampled from a torus.
The barcodes extending to infinity are the ones corresponding to the true "data", the rest can be counted as "noise".

2.8 A Visualization of Noise and Data

To visualize the concept of noise and data in a filtration, the following figures will be extremely useful to gain insight into this concept from *Figure 2.9* and *Figure 2.10*.

In *Figure 2.9*, it can be observed that particular distance parameter, there are three "holes". One large one and two smaller ones.

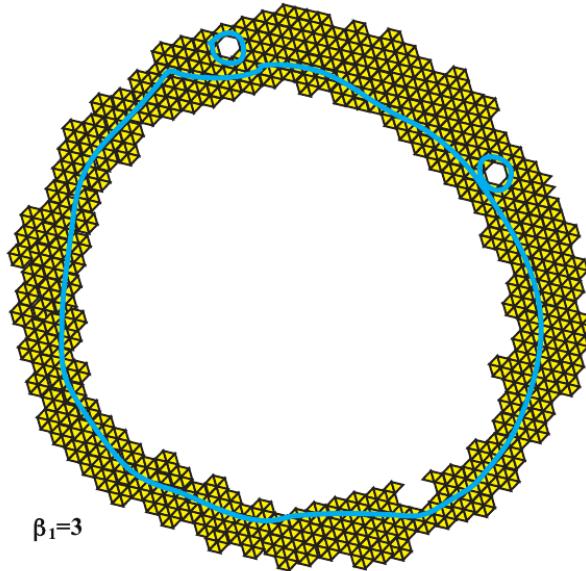


Figure 2.9: Data Points sampled from a torus.[7]

The moment the distance parameter is increased a little, the case shown in *Figure 2.10* is achieved, where the smaller holes are now filled up by simplices, yet a new hole appears on the lower right side of the figure. If the distance parameter is increased further, this "hole" will also cease to exist.

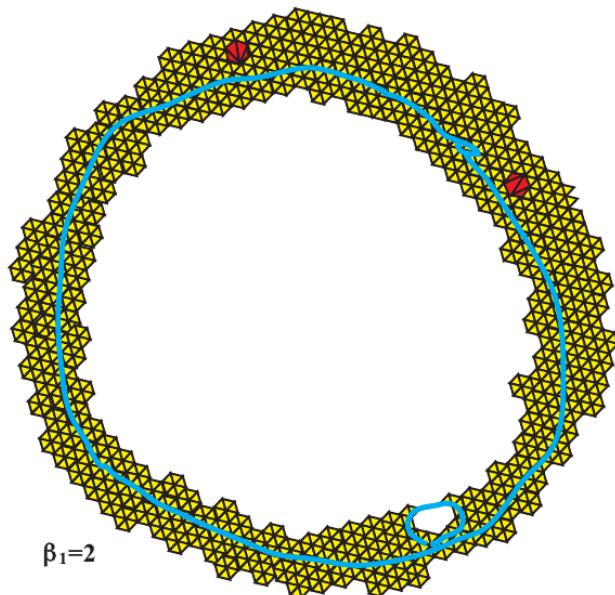


Figure 2.10: Data Points sampled from a torus.[7]

Hence, the only "hole" that persists is the large hole in the centre of the data. Thus it can be concluded that this large hole alone corresponds to data, while the others correspond to noise or artefacts. A large amount of artefacts could also mean that the data that has chosen is noisy or badly sampled.

2A Appendix - Homology

2A.1 Graphs

A *graph* is a pair of sets (\mathbf{V}, \mathbf{E}) where \mathbf{V} is a finite set called the set of *vertices* and \mathbf{E} is a set of unordered 2-element subsets of \mathbf{V} , called the set of *edges*. We view the *edges* as a set of connections between the nodes. \mathbf{E} is a *multiset*, thus it's elements can occur more than once so that every element has a *multiplicity*.[8]

Example

Here are some examples of a few graphs :

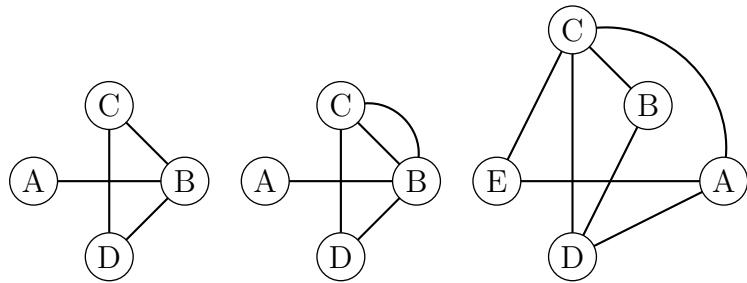


Figure 2.11: Examples of Graphs

2A.2 Nerve Theorem

The nerve is the homotopy equivalent to the original space if the sets in the covering are all iterated intersections are empty or contractible.[9]

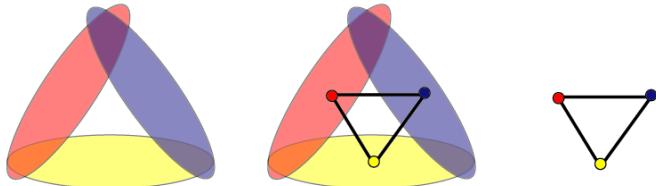


Figure 2.12: Constructing the nerve of an open good cover containing 3 sets in the plane.[10]

2A.3 Point Cloud

The set of coordinates in n-dimensional Euclidean space that correspond to the data set. In this document, we shall deal only with a 2-dimensional data set and hence the point cloud will consist only of a pair of coordinates namely the abscissa and the ordinate i.e (x, y) pairs.

2A.4 Group Kernel

The Kernel of a group homomorphism $f : G \Rightarrow G'$ is the set of all the elements of G which are mapped to the identity element of G' . In the context of this document, the identity element is $\mathbf{0}$. Thus the Kernel can also be understood as an equivalent to the well-known concept of "null-space" in linear algebra. In fact, the null-space is a limiting case of a kernel.[11] With respect to the scope of discussion within this document, the kernel of a map can simply be understood as the set of those elements that get mapped to 0 under that particular operation.

2A.5 The Boundary ' ∂ ' Operator

In algebraic topology, a k -chain is a formal linear combination of k -cells in a cell complex. The ∂ operator can be explained as an operator that acts on a k -chain to give a $k - 1$ chain. It is important to note that the boundary of a simplex is not a simplex, rather a series of simplices with coefficients with '1' or '-1'. These maps from k chains to $k - 1$ chains are formed by the boundary operator.[5]

Boundaries (\mathbb{B}) are a subset of cycles (\mathbb{Z}). Further, cycles (\mathbb{Z}) are a subset of all chains(\mathbb{C}) of that dimension. Therefore, $\mathbb{B}_n \subset \mathbb{Z}_n \subset \mathbb{C}_n$.

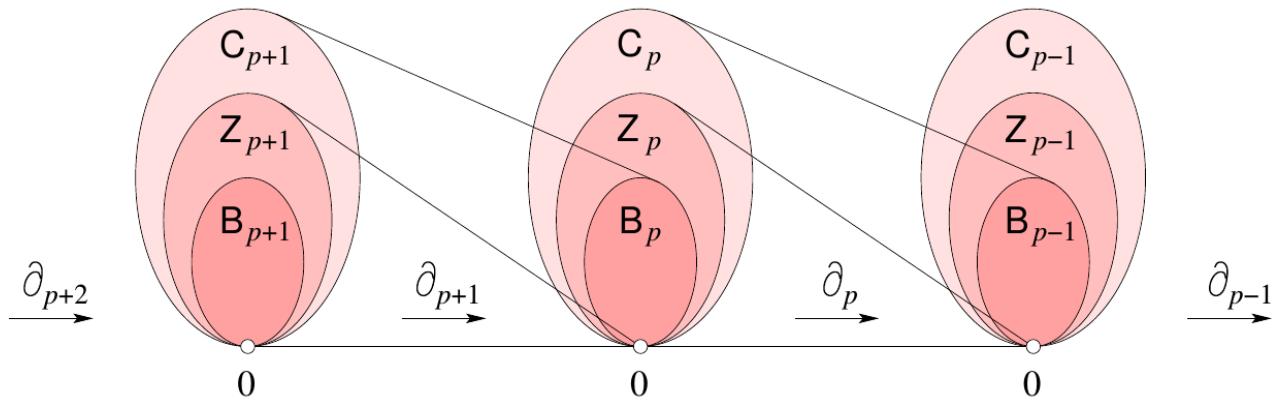


Figure 2.13: Chain Complex - linear chains, cycles and boundaries. A visualization of the boundary operator[5]

2A.6 Free Abelian Group

A group is "Abelian" if the group operation is commutative over all the group elements. A free Abelian group is an Abelian group with a "basis". Basis comprises of a subset of the group such that every element of the group can be represented by an integral linear combination of basis elements and is unique.

CHAPTER 3

Nonlinear Dynamics

3.1 Introduction

Dynamics is the study of the evolution of systems with time. The concepts of dynamics are applied in various fields including classical mechanics, chemical kinetics, population biology and network analysis to name a few. When a system exhibits linear behaviour, it can be solved easily using analytical methods. A nonlinear system on the other hand, cannot be solved analytically. In the age of computers, it can be solved with good precision by using numerical methods.

Nonlinear dynamics involves solving differential equations. Numerical methods used by computers involve approximations. Given the initial conditions, the solver then finds the phase space trajectory after a particular elapsed time.

Let us consider an n-dimensional autonomous dynamical system such that

$$\dot{\vec{X}} = f(X) \quad (3.1)$$

$$\vec{X} = (x_1, x_2, x_3, \dots x_n) , \quad x \in \mathbb{R}^n \quad (3.2)$$

$$\vec{f} = (f_1, f_2, f_3, \dots f_n) \quad (3.3)$$

It is possible to have a non-autonomous system as well. Such a system would exhibit

explicit time dependence and would be described as follows-

$$\dot{\vec{X}} = f(\vec{X}, t) \quad (3.4)$$

The quantity $\dot{\vec{X}}$ is the velocity in phase space and $f(\vec{X})$ specifies locally at each point the direction and magnitude in which the vector \vec{X} will change. Thus given certain initial conditions, it is possible to predict the phase space trajectory after an infinitesimal time dt . Thus from this predicted point it is possible to predict the subsequent point in phase space by further predicting the phase space trajectory after another dt time interval. Upon repeating this process, it is possible to plot the phase space trajectory by solving locally using approximations.

It is important to note that the phase space trajectory calculated by this method is an approximate solution to the nonlinear system. There shall exist an error in each step and in a very highly nonlinear system such errors may multiply with each step. Though such computational challenges exist, for simpler systems, modern computers can calculate phase space trajectories with good accuracy.

3.2 Duffing Oscillator

The Duffing oscillator is a second order non-linear oscillator. The duffing equation consists of both damping and driving forces and thus is appropriate for a system that consists of both damping and driving forces. [12]

$$\ddot{x} + \delta(\dot{x}) - \alpha x(t) + \beta x^3(t) = \gamma \cos(\omega t) \quad (3.5)$$

δ - controls the amount of damping.

α - controls the linear stiffness.

β - controls the amount of non-linearity in the restoring force; if $\beta = 0$, the Duffing equation describes a damped and driven simple harmonic oscillator,

γ - is the amplitude of the periodic driving force; if $\gamma = 0$, the system is without a

driving force.

ω - is the angular frequency of the periodic driving force.

The parameters chosen for our study are -

$$\delta = 0.3$$

$$\alpha = -1$$

$$\beta = 1$$

$$\omega = 1.2$$

$$\gamma = [0.35, 0.38]$$

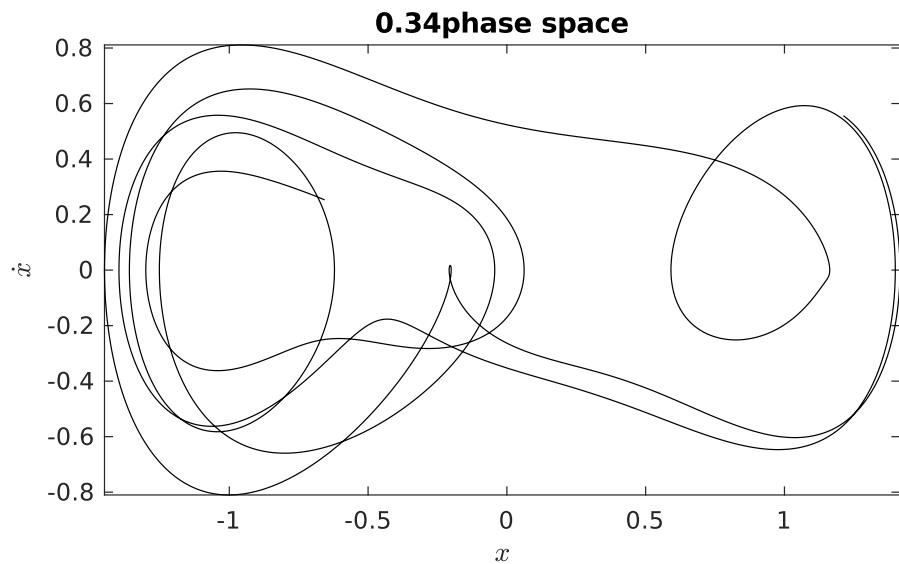
Upon solving, the duffing equation we arrive at:

$$\dot{x} = y \quad (3.6)$$

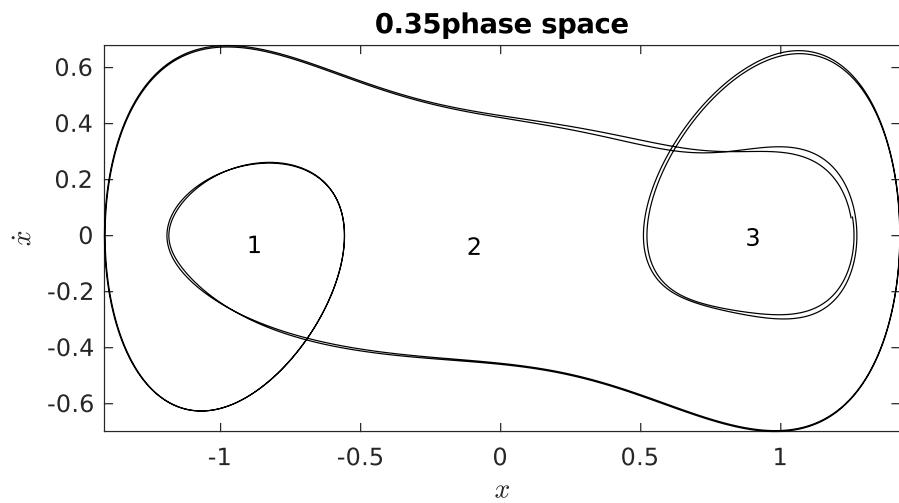
An interesting and important consequence of the above condition is that the phase portrait \dot{x} vs x is equivalent to y vs x . Hence, this introduces a certain versatility and these two terms can be used interchangeably.

3.3 Phase-space Plots

An analysis of the phase space plots of the duffing oscillator corresponding to various values of γ , while keeping all the other values constant.

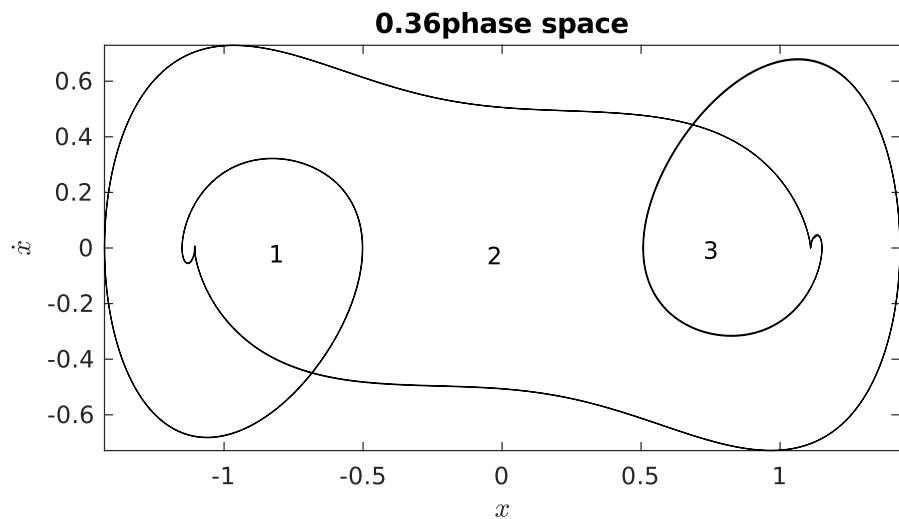


(a) Phase space plot for $\gamma = 0.34$, exhibiting chaotic behaviour.

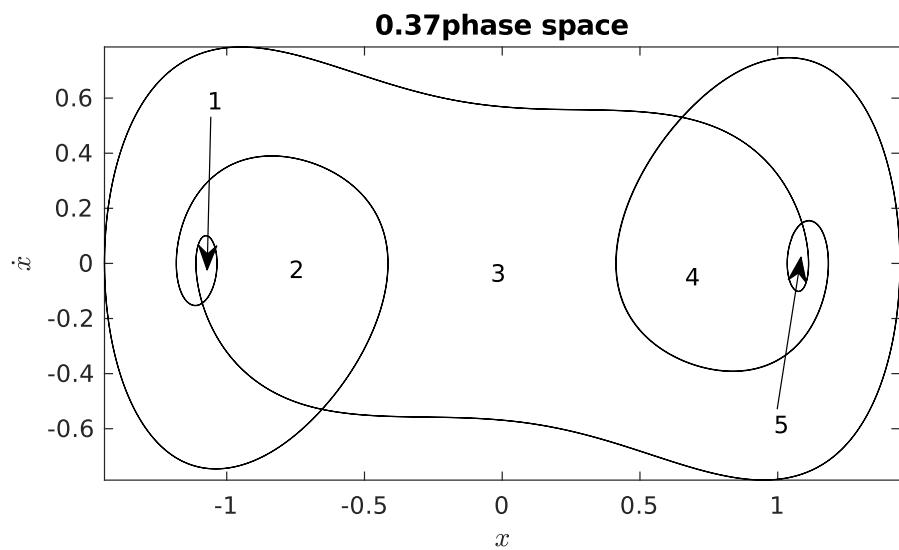


(b) Phase space plot for $\gamma = 0.35$, exhibiting period - 3 behaviour .

Figure 3.1: Phase space plots

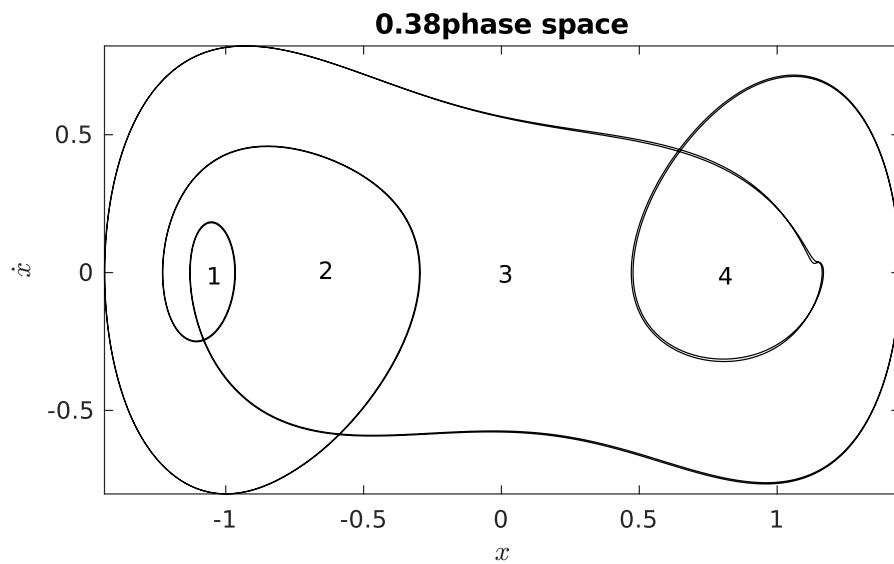


(a) Phase space plot for $\gamma = 0.36$, exhibiting period - 3 behaviour.

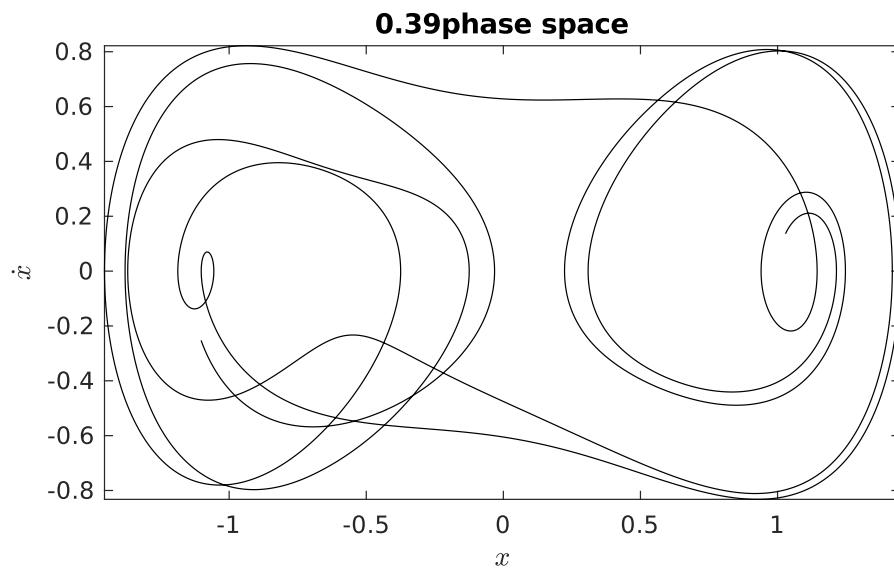


(b) Phase space plot for $\gamma = 0.37$, exhibiting period - 5 behaviour.

Figure 3.2: Phase space plots



(a) Phase space plot for $\gamma = 0.38$, exhibiting period - 4 behaviour.



(b) Phase space plot for $\gamma = 0.39$, exhibiting chaotic behaviour.

Figure 3.3: Phase space plots

3.4 Periodicity

The periodic behaviour of these systems can be linked to the betti numbers. The calculation is simple and straightforward, the period of oscillation of the system is equal to the betti number of the structure formed in the phase space portrait. This follows from the fact that, the periodicity is equal to the number of loops in the phase space portrait. This concept can be affirmed by comparing the time series with the phase space portrait of the system.

As we can observe from the above plots, $\gamma = 0.34$ and $\gamma = 0.39$ are both chaotic in nature. $\gamma = 0.35$ is almost periodic in nature and can be approximated to period - 3 behaviour, such a case is a good example to study for a system that is moving from chaos into periodicity. $\gamma = 0.36$ exhibits period - 3 behaviour. $\gamma = 0.37$ exhibits period - 5 behaviour while $\gamma = 0.38$ shows period - 4 behaviour.

Note- It is important to note that all the phase space plots shown above are projections on the 2-D plane.

CHAPTER 4

Application of Persistent Homology

4.1 Introduction

Since the periodicity of a system can be determined by the betti numbers, we shall use persistent homology to extract the underlying topology from a point cloud dataset that has been generated by extracting points from the phase space portrait of a physical system. The process is described in detail below.

4.2 Generating the point cloud dataset

The Duffing equation is solved by using a differential equation solver and then equispaced points are extracted from the phase portrait of the duffing oscillator. This forms the point cloud dataset in 2 dimensions. It is important to skip the transients in a system and thus the plotting was started after skipping the first $\frac{2\pi}{1000} \times 10000 = 52.36$ seconds. This done to eliminate errors that may arise due to transients in the oscillator.

For $\gamma = 0.35, 0.36, 0.37$ and 0.38 , 100, 200, 303 and 400 equispaced points were obtained from the respective phase space plots to form the point cloud dataset, as shown in *Figure 4.1*

4.3 Input to Javaplex

The points in the dataset are then passed onto a javaplex[13] stream. Javaplex then creates a VR complex and then performs Delaunay triangulation for each value of the distance parameter. Also, the homology group generators are calculated for each value of the distance parameter. Using the generated data the birth and death times of all the homology group generators are extracted and then the lifetimes of the homology group generators calculated.

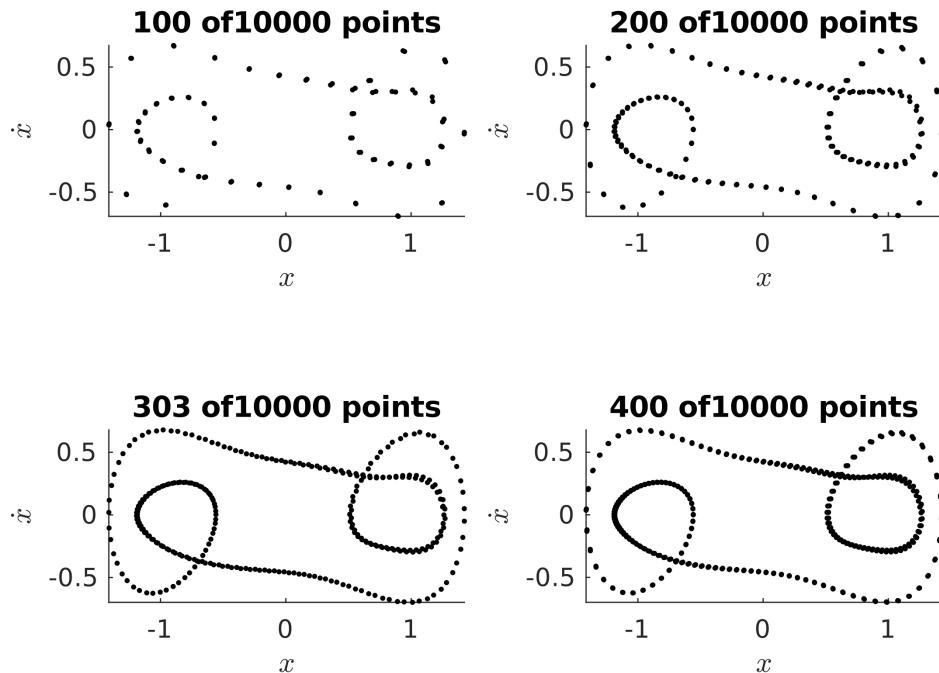
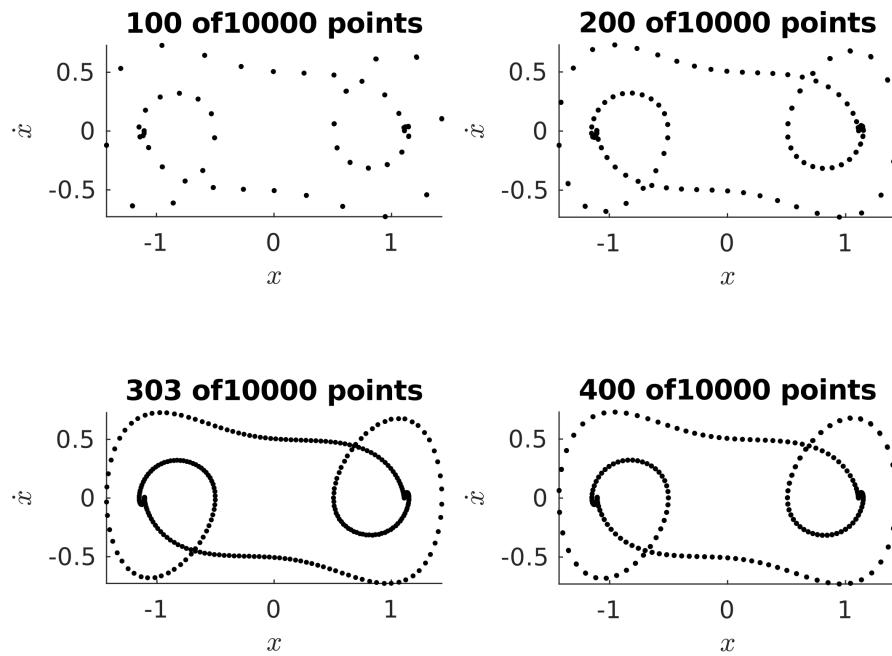
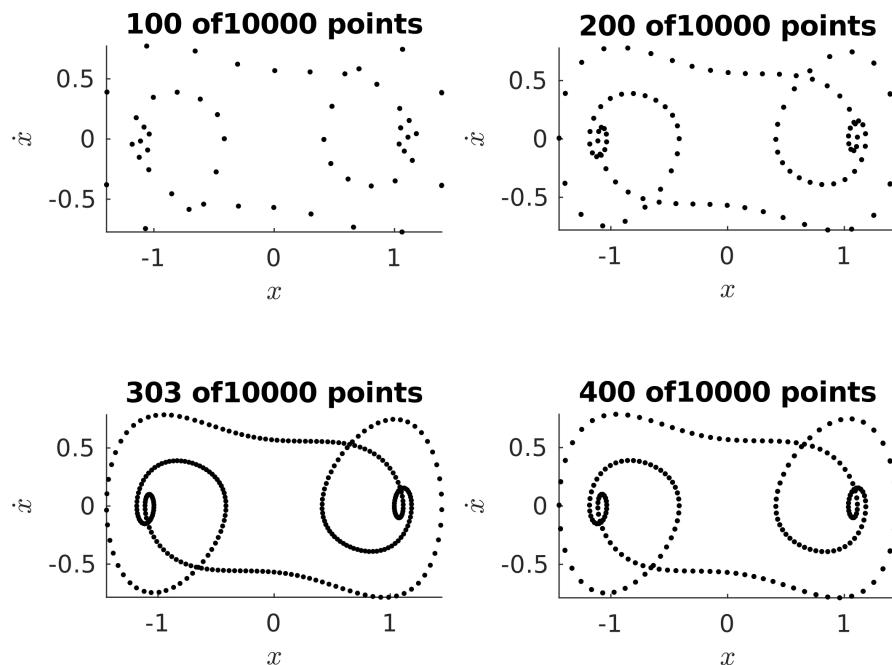


Figure 4.1: Points extracted from the phase space plot for $\gamma = 0.35$.

(a) Points extracted from the phase space plot for $\gamma = 0.36$.(b) Points extracted from the phase space plot for $\gamma = 0.37$.Figure 4.2: Points Passed to Javaplex, $\gamma = 0.35, 0.36$

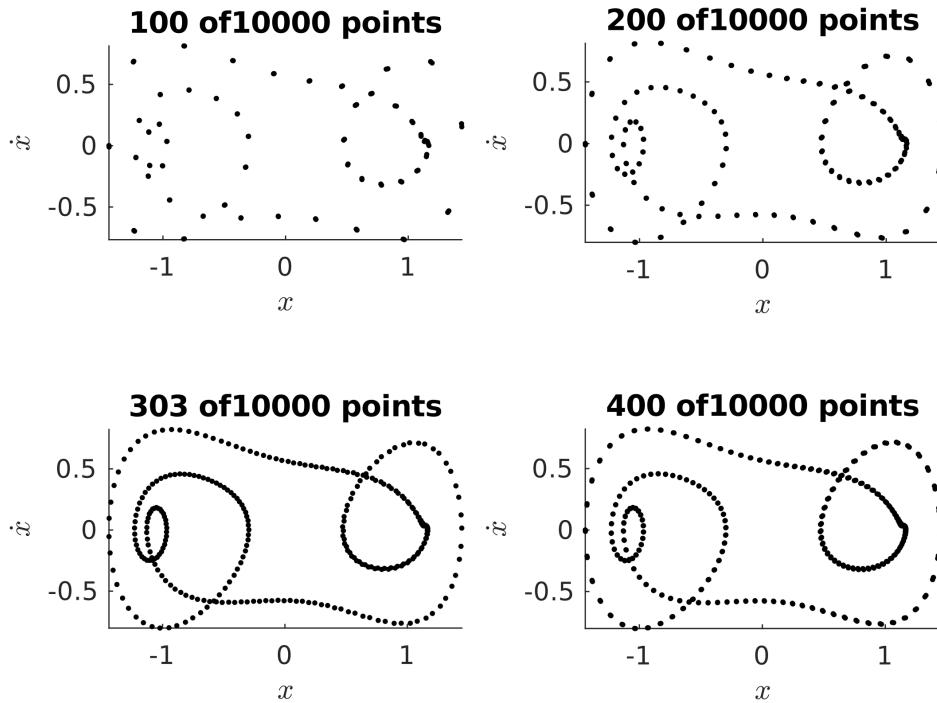


Figure 4.3: Points extracted from the phase space plot for $\gamma = 0.38$.

4.4 Output of Javaplex

The Lifetimes of the homology group generators calculated by the Javaplex plug-in contain both data and noise. Drawing from the idea of persistence, it is easy to deduce that the homology group generators with short lifetimes correspond to noise while the ones that are longer correspond to data. It is a challenge to differentiate between noise and data. Defining a threshold for noise and data is important to be able to interpret the lifetimes.

It is logical to turn to statistical analysis for defining a threshold to differentiate between noise and data in the system. Since the data requires a binary classification, ROC (Receiver Operating Characteristic) is a good option to turn to.

4.5 Barcode Plots

The barcode plots generated by Javaplex have been displayed below. It is possible to correlate the period of the system with the lifetimes of the homology group generators. As can be noticed in *Figure 4.4* there are three long lines in the barcode plot and the period of the system is equal to 3. Similarly, all the graphs can be analyzed.

Barcode Plots for $\gamma = 0.35$, Period - 3 Behaviour

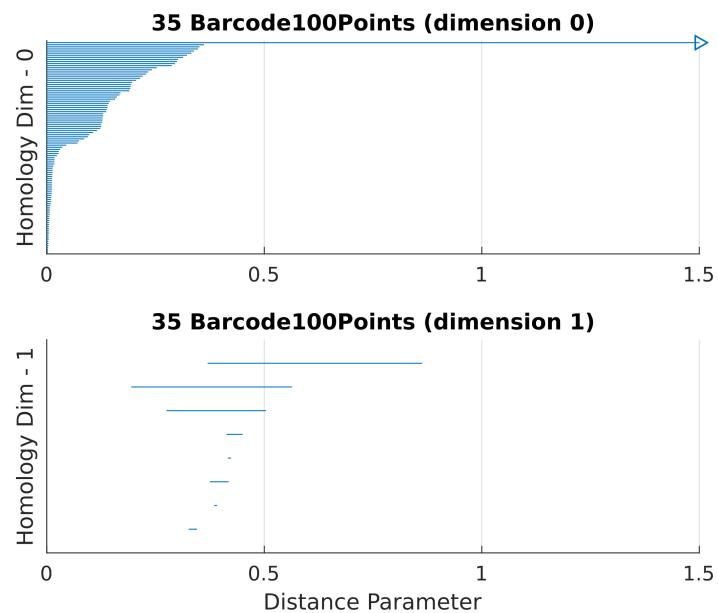
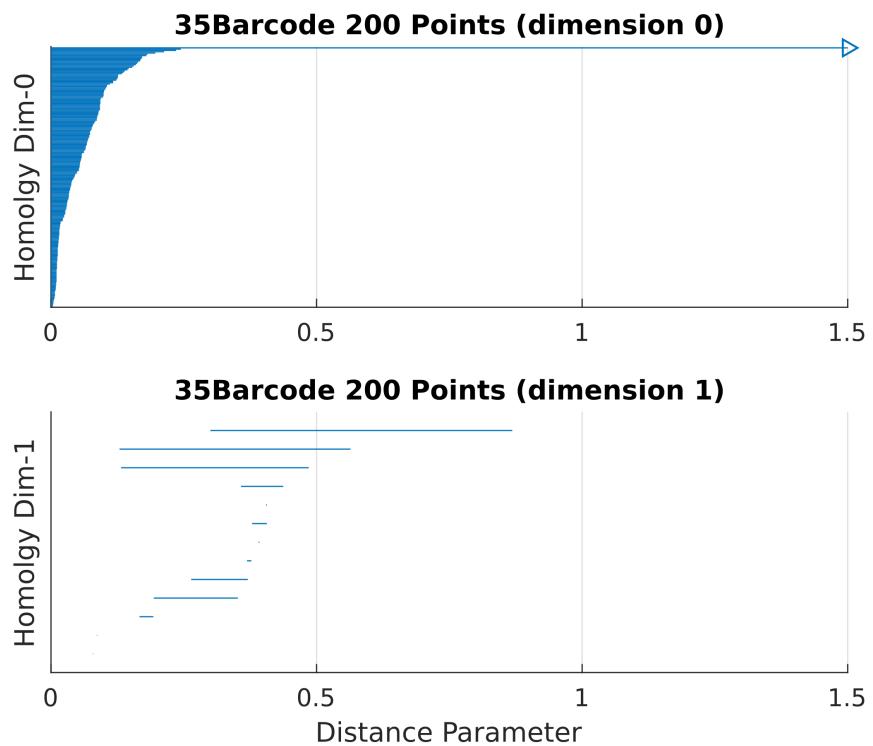
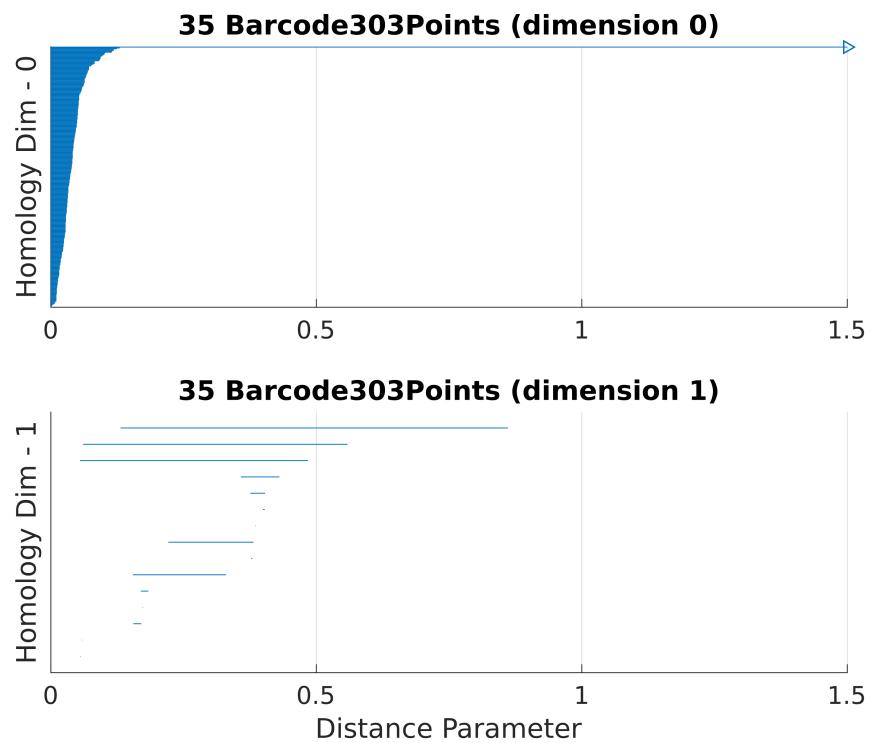


Figure 4.4: Barcode plot for $\gamma = 0.35$ using 100 data points. Period - 3 Behaviour



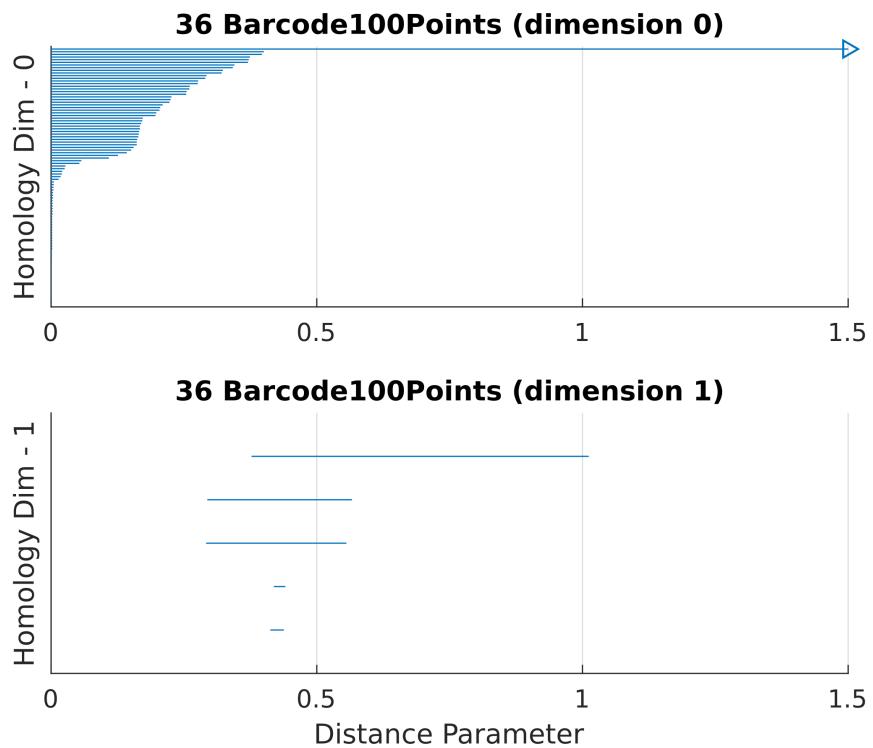
(a) Barcode plot for $\gamma = 0.35$ using 200 data points. Period - 3 Behaviour



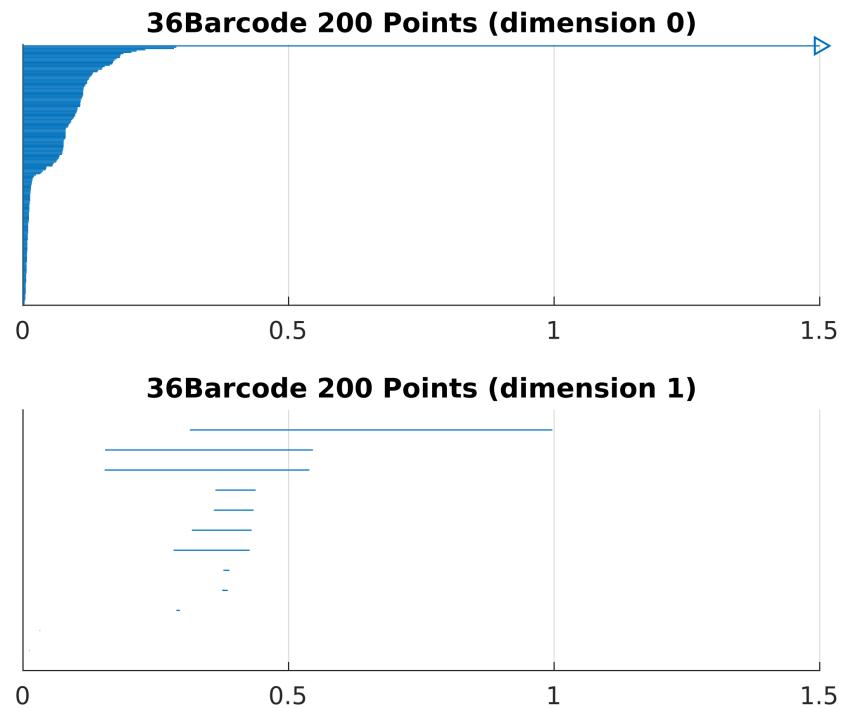
(b) Barcode plot for $\gamma = 0.35$ using 303 data points. Period - 3 Behaviour

Figure 4.5

Barcode Plots for $\gamma = 0.36$, Period - 3 Behaviour



(a) Barcode plot for $\gamma = 0.36$ using 100 data points. Period - 3 Behaviour



(b) Barcode plot for $\gamma = 0.36$ using 200 data points. Period - 3 Behaviour

Figure 4.6

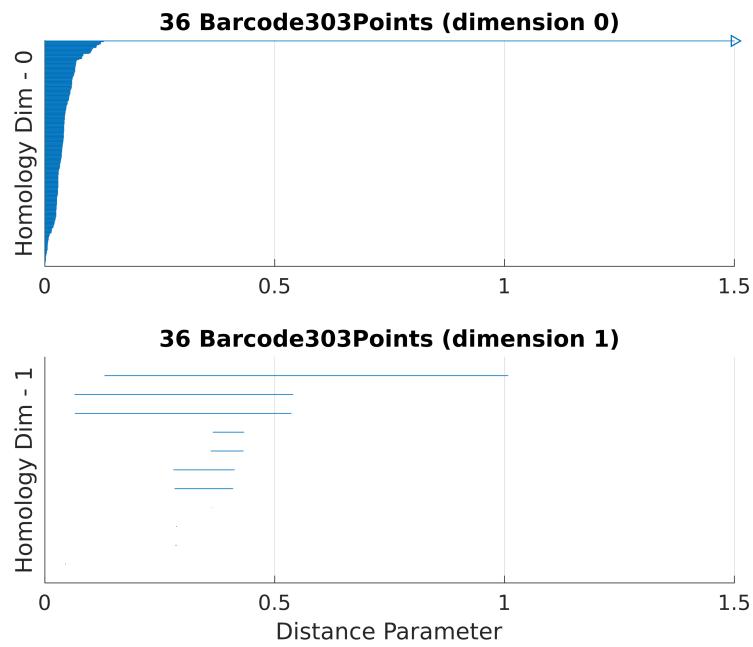


Figure 4.7: Barcode plot for $\gamma = 0.36$ using 303 data points. Period - 3 Behaviour

Barcode Plots for $\gamma = 0.37$, Period - 5 Behaviour

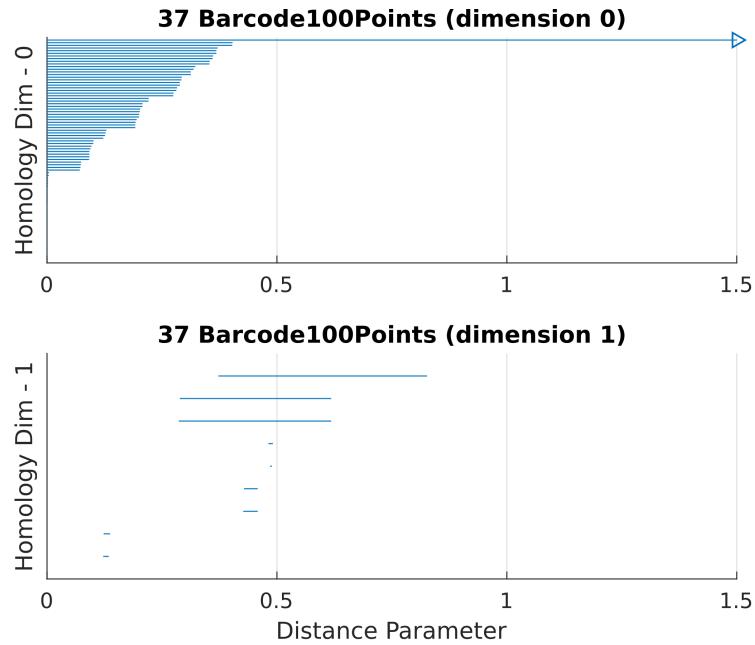
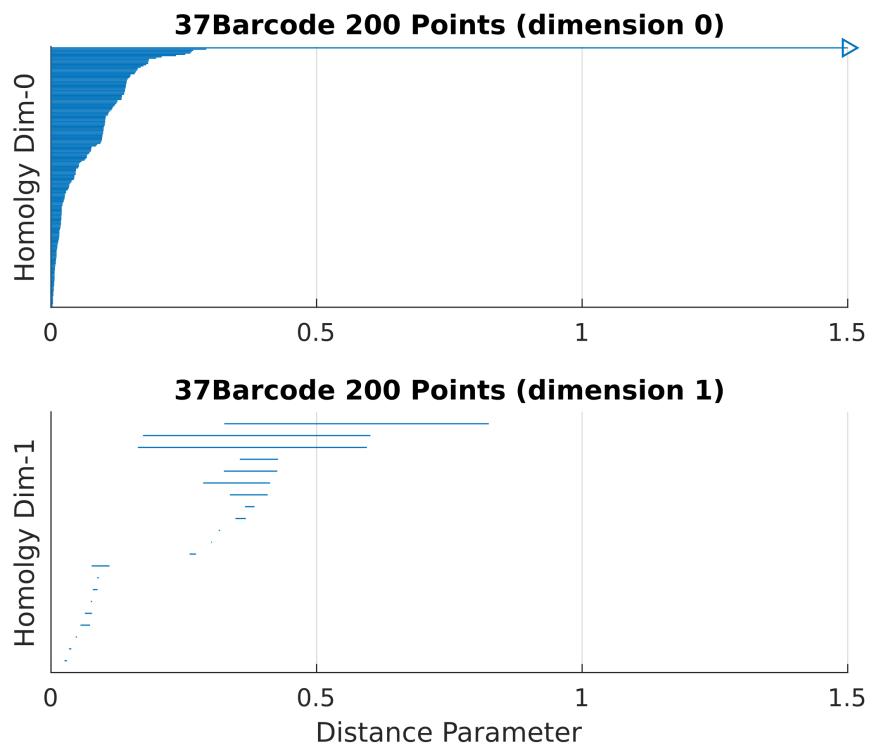
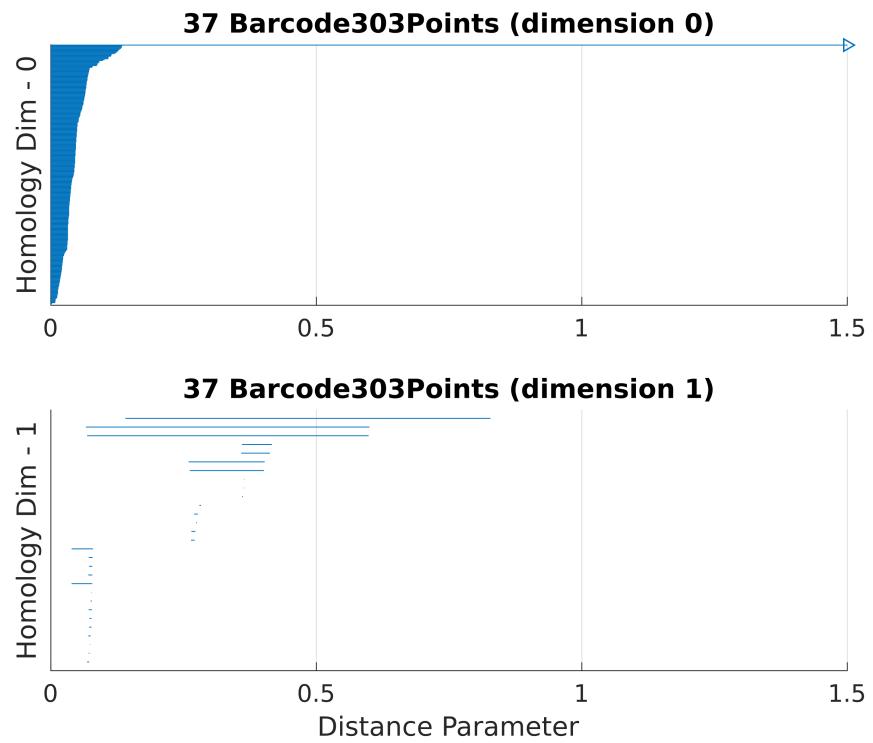


Figure 4.8: Barcode plot for $\gamma = 0.37$ using 100 data points. Period - 5 Behaviour



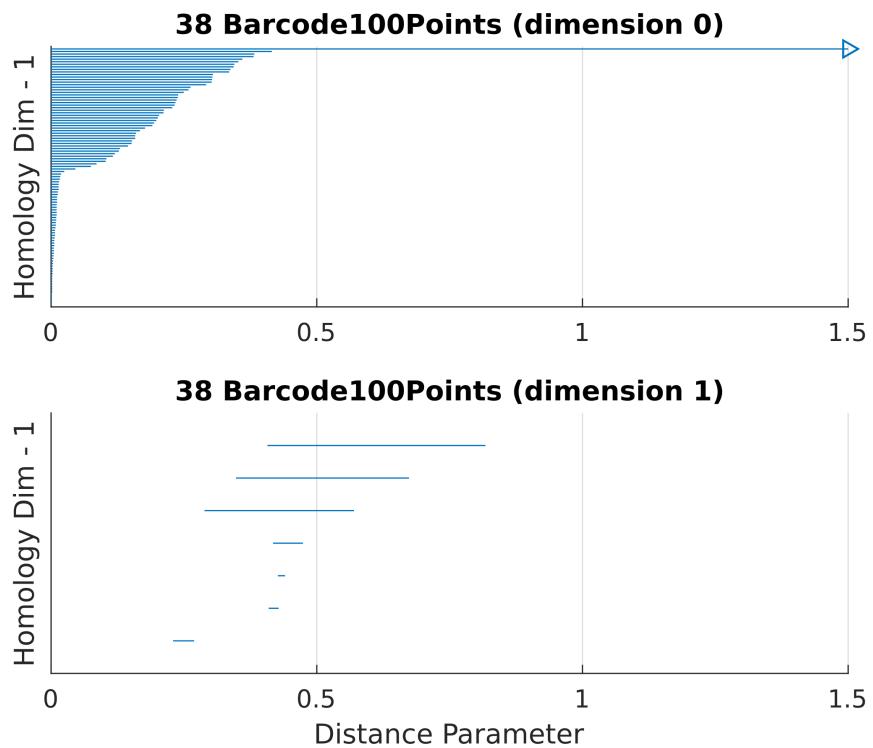
(a) Barcode plot for $\gamma = 0.37$ using 200 data points. Period - 5 Behaviour



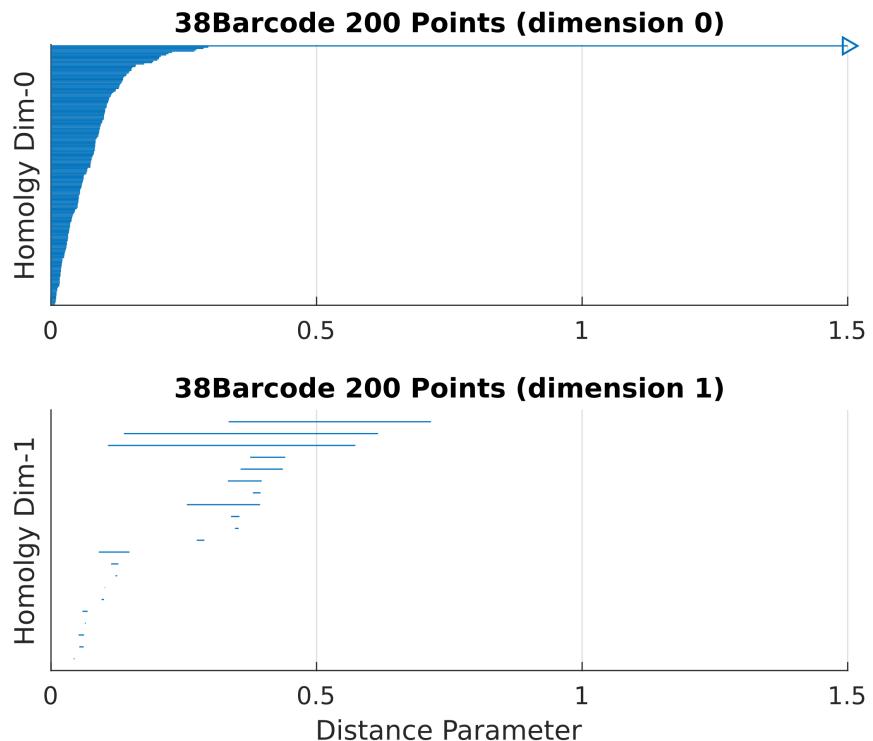
(b) Barcode plot for $\gamma = 0.37$ using 303 data points. Period - 5 Behaviour

Figure 4.9

Barcode Plots for $\gamma = 0.38$, Period - 4 Behaviour



(a) Barcode plot for $\gamma = 0.38$ using 100 data points. Period - 4 Behaviour



(b) Barcode plot for $\gamma = 0.38$ using 200 data points. Period - 4 Behaviour

Figure 4.10

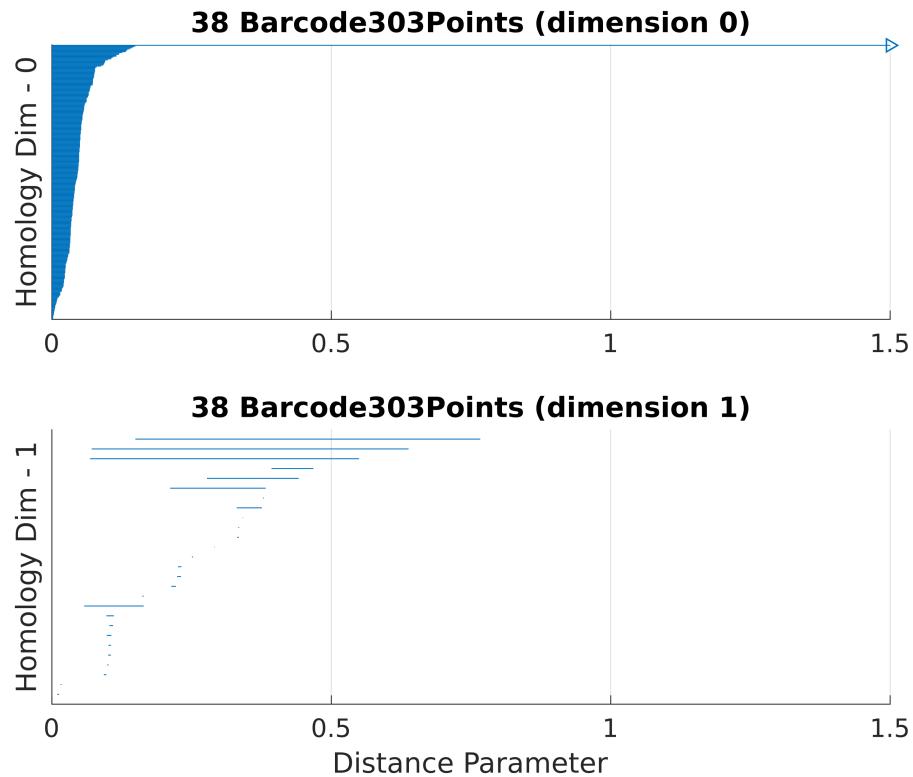


Figure 4.11: Barcode plot for $\gamma = 0.38$ using 303 data points. Period - 4 Behaviour

4.6 Birth Time, Death Time and Lifetime Data

The Following tables contain the Birth Time, Death Time and Lifetimes for $\gamma = 0.35, 0.36, 0.37, 0.38$ for iterations with 100, 200, 303 and 400 data points. *The lifetimes that are marked in bold and encircled are the ones that correspond to data, while the others correspond to noise.*

4.6.1 Lifetimes for $\gamma = 0.35$

Birth Time	Death Time	Lifetime
0.331	0.341	0.010
0.354	0.399	0.045
0.373	0.420	0.048
0.415	0.434	0.020
0.407	0.455	0.048
0.245	0.514	0.269
0.195	0.562	0.367
0.377	0.858	0.481

Table 4.1: Birth times, death times and lifetimes for $\gamma = 0.35$ using 200 points.

Birth Time	Death Time	Lifetime
0.079	0.081	0.003
0.087	0.087	0.001
0.251	0.344	0.093
0.371	0.377	0.007
0.387	0.388	0.001
0.284	0.392	0.108
0.401	0.402	0.001
0.375	0.403	0.028
0.359	0.440	0.081
0.141	0.491	0.350
0.131	0.564	0.434
0.309	0.864	0.555

Table 4.2: Birth times, death times and lifetimes for $\gamma = 0.35$ using 200 points.

Birth Time	Death Time	Lifetime
0.053	0.053	0.000
0.053	0.054	0.001
0.053	0.054	0.001
0.204	0.208	0.004
0.239	0.243	0.004
0.189	0.333	0.145
0.362	0.363	0.001
0.382	0.385	0.003
0.227	0.387	0.160
0.393	0.396	0.003
0.375	0.401	0.027
0.360	0.437	0.077
0.094	0.485	0.392
0.084	0.558	0.475
0.204	0.862	0.658

Table 4.3: Birth times, death times and lifetimes for $\gamma = 0.35$ using 303 points.

Birth Time	Death Time	Lifetime
0.046	0.047	0.002
0.057	0.058	0.001
0.083	0.084	0.001
0.173	0.184	0.011
0.164	0.185	0.022
0.234	0.238	0.004
0.162	0.329	0.167
0.360	0.362	0.002
0.378	0.380	0.002
0.379	0.381	0.002
0.223	0.384	0.161
0.399	0.401	0.003
0.375	0.402	0.027
0.359	0.432	0.073
0.074	0.485	0.411
0.066	0.557	0.492
0.161	0.861	0.700

Table 4.4: Birth times, death times and lifetimes for $\gamma = 0.35$ using 400 points.

4.6.2 Lifetimes for $\gamma = 0.36$

Birth Time	Death Time	Lifetime
0.413	0.438	0.026
0.419	0.441	0.022
0.292	0.556	0.264
0.294	0.566	0.272
0.377	1.012	0.635

Table 4.5: Birth times, death times and lifetimes for $\gamma = 0.36$ using 100 points.

Birth Time	Death Time	Lifetime
0.378	0.387	0.009
0.381	0.391	0.010
0.396	0.405	0.009
0.395	0.413	0.018
0.329	0.421	0.092
0.331	0.430	0.100
0.362	0.439	0.077
0.365	0.442	0.077
0.157	0.554	0.397
0.160	0.554	0.395
0.316	1.011	0.695

Table 4.6: Birth times, death times and lifetimes for $\gamma = 0.36$ using 200 points.

Birth Time	Death Time	Lifetime
0.041	0.041	0.001
0.045	0.047	0.002
0.293	0.304	0.011
0.365	0.368	0.004
0.374	0.377	0.003
0.364	0.434	0.071
0.366	0.437	0.071
0.107	0.538	0.431
0.113	0.551	0.438
0.233	1.009	0.776

Table 4.7: Birth times, death times and lifetimes for $\gamma = 0.36$ using 303 points.

Birth Time	Death Time	Lifetime
0.008	0.008	0.000
0.009	0.010	0.000
0.009	0.010	0.002
0.010	0.011	0.001
0.011	0.012	0.001
0.010	0.012	0.003
0.285	0.290	0.005
0.287	0.291	0.004
0.362	0.363	0.001
0.365	0.367	0.002
0.283	0.409	0.127
0.282	0.411	0.129
0.359	0.435	0.076
0.362	0.438	0.076
0.085	0.534	0.449
0.082	0.535	0.454
0.176	0.998	0.822

Table 4.8: Birth times, death times and lifetimes for $\gamma = 0.36$ using 400 points.

4.6.3 Lifetimes for $\gamma = 0.37$

Birth Time	Death Time	Lifetime
0.124	0.137	0.014
0.124	0.137	0.014
0.429	0.460	0.031
0.429	0.460	0.031
0.486	0.491	0.005
0.487	0.491	0.004
0.289	0.619	0.330
0.289	0.619	0.330
0.373	0.828	0.455

Table 4.9: Birth times, death times and lifetimes for $\gamma = 0.37$ using 100 points.

Birth Time	Death Time	Lifetime
0.080	0.092	0.013
0.080	0.093	0.013
0.079	0.112	0.033
0.079	0.115	0.037
0.367	0.380	0.013
0.368	0.381	0.013
0.363	0.427	0.064
0.364	0.429	0.065
0.350	0.430	0.080
0.350	0.433	0.083
0.182	0.606	0.425
0.180	0.607	0.427
0.323	0.827	0.504

Table 4.10: Birth times, death times and lifetimes for $\gamma = 0.37$ using 200 points.

Birth Time	Death Time	Lifetime
0.071	0.085	0.014
0.077	0.090	0.014
0.072	0.093	0.021
0.071	0.093	0.023
0.072	0.095	0.024
0.276	0.299	0.023
0.283	0.301	0.018
0.295	0.314	0.019
0.365	0.366	0.001
0.365	0.370	0.005
0.374	0.378	0.004
0.383	0.384	0.001
0.382	0.385	0.003
0.396	0.399	0.003
0.264	0.407	0.143
0.266	0.415	0.149
0.360	0.417	0.057
0.360	0.419	0.059
0.123	0.601	0.478
0.122	0.605	0.483
0.261	0.828	0.568

Table 4.11: Birth times, death times and lifetimes for $\gamma = 0.37$ using 303 points.

Birth Time	Death Time	Lifetime
0.075	0.080	0.005
0.057	0.081	0.024
0.076	0.081	0.005
0.059	0.083	0.024
0.079	0.084	0.005
0.080	0.084	0.004
0.280	0.285	0.005
0.275	0.286	0.012
0.281	0.287	0.006
0.277	0.289	0.013
0.361	0.363	0.002
0.362	0.364	0.002
0.367	0.368	0.001
0.368	0.369	0.001
0.372	0.373	0.001
0.259	0.407	0.148
0.261	0.407	0.147
0.359	0.415	0.056
0.360	0.416	0.056
0.099	0.598	0.500
0.097	0.599	0.502
0.200	0.827	0.627

Table 4.12: Birth times, death times and lifetimes for $\gamma = 0.37$ using 400 points.

4.6.4 Lifetimes for $\gamma = 0.38$

Birth Time	Death Time	Lifetime
0.231	0.267	0.036
0.407	0.424	0.017
0.433	0.434	0.001
0.424	0.437	0.013
0.419	0.473	0.054
0.289	0.564	0.276
0.343	0.673	0.330
0.407	0.821	0.414

Table 4.13: My caption

Table 4.14: Birth times, death times and lifetimes for $\gamma = 0.38$ using 100 points.

Birth Time	Death Time	Lifetime
0.084	0.086	0.002
0.125	0.166	0.042
0.336	0.357	0.022
0.344	0.362	0.018
0.335	0.390	0.055
0.413	0.414	0.002
0.309	0.422	0.113
0.318	0.446	0.128
0.393	0.466	0.073
0.154	0.565	0.411
0.187	0.653	0.466
0.315	0.773	0.458

Table 4.15: Birth times, death times and lifetimes for $\gamma = 0.38$ using 200 points.

Birth Time	Death Time	Lifetime
0.011	0.011	0.000
0.098	0.127	0.029
0.103	0.130	0.027
0.100	0.130	0.031
0.157	0.161	0.004
0.097	0.167	0.071
0.324	0.326	0.002
0.334	0.345	0.011
0.349	0.352	0.003
0.332	0.383	0.052
0.259	0.397	0.138
0.277	0.454	0.177
0.393	0.464	0.071
0.117	0.554	0.437
0.131	0.640	0.509
0.253	0.766	0.513

Table 4.16: Birth times, death times and lifetimes for $\gamma = 0.38$ using 303 points.

Birth Time	Death Time	Lifetime
0.010	0.012	0.002
0.082	0.083	0.001
0.096	0.108	0.012
0.099	0.112	0.013
0.100	0.113	0.013
0.099	0.117	0.018
0.108	0.118	0.010
0.161	0.166	0.005
0.081	0.166	0.086
0.229	0.236	0.007
0.223	0.250	0.027
0.256	0.271	0.015
0.334	0.335	0.001
0.336	0.340	0.004
0.343	0.346	0.004
0.370	0.375	0.005
0.331	0.378	0.047
0.211	0.388	0.177
0.398	0.399	0.001
0.278	0.446	0.168
0.393	0.466	0.073
0.080	0.551	0.471
0.099	0.642	0.543
0.203	0.767	0.565

Table 4.17: Birth times, death times and lifetimes for $\gamma = 0.38$ using 303 points.

4.6.5 Interpretation of Lifetimes

From the above data, it can be observed that it is necessary to define a threshold to differentiate between noise and data. This is important because to be able to characterize the periodic behaviour of a system, it is necessary to draw correlations between the lifetimes and the betti numbers of the topology of data.

CHAPTER 5

Statistical Analysis

5.1 Motivation

Persistent homology places upon the researcher a burden to interpret an array of values of the lifetimes of Homology group generators. It is necessary to segregate this array of results into noise and data. It is a well-established fact that the longer lines correspond to data, while the shorter ones correspond to noise. It is important to pick a threshold for each system so that noise and data can be segregated. Thus a statistical analysis of the lifetimes can provide insight about how to pick this threshold value.

5.2 Receiver Operating Characteristic

5.2.1 Introduction

Lifetime-values need to be segregated into two categories. They are either "data" or "noise". Thus a binary statistical test is required to categorize the lifetime-values into data or noise.

Receiver Operating Characteristic is one such statistical tool that is designed for binary testing on the basis of a single variable. ROC is widely used in medical tests to help improve diagnosis. For example in a test for glaucoma, while measuring the fluid pressure

in a patient's eyes, we know that people with eye pressure values below 10 are definitely healthy. Similarly, all those with values above 40 are definitely tested positive for glaucoma. For cases with values between 10 and 40, it is not possible to predict the result with 100 percent accuracy. ROC plays a vital role in diagnosing the condition of such patients.

In an ROC analysis, we end up with four categories -

True Positives (TP) - When the condition is positive and the test result is also positive.

True Negatives (TN) - When the condition is negative and the test result is also negative.

False Positives (FP)- When the condition is negative but the test result is positive.

False Negatives (FN)- When the condition is positive but the test result is negative.

		True Condition	
		Condition Negative	Condition Positive
Predicted Condition	Predicted Positive	True Positive	False Positive
	Predicted Negative	False Negative	True Negative

Table 5.1: ROC Categorization

These four categories can be represented on a graph with the dependent variable on the $x - axis$ and the density of states on the $y - axis$.

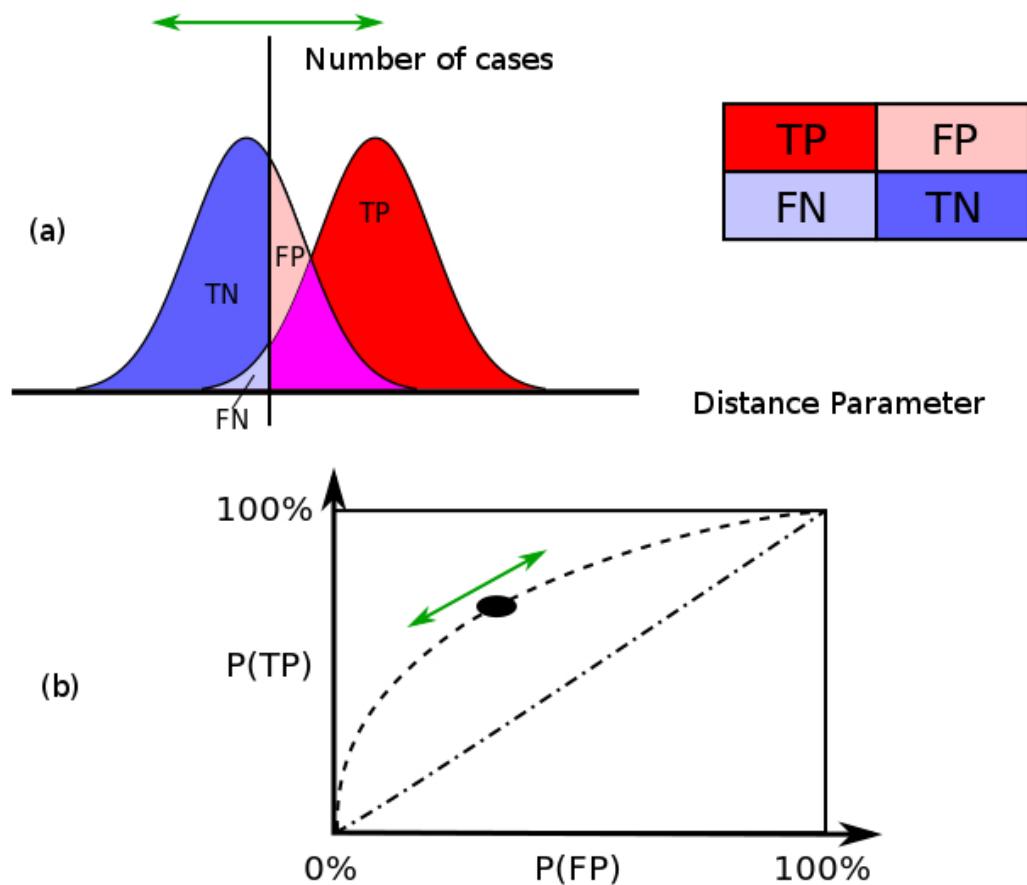


Figure 5.1: ROC Curve

Bayesian Probabilistic Measures

Bayesian probabilistic measures are conditional probabilistic measures. Such a measure is used to find the probability of an event, given that another event has occurred/will definitely occur. Sensitivity and Specificity are two such measures. Given that the condition is positive, the probability that the test result is positive is the sensitivity of the test. Similarly, given that the condition is negative, the probability that the test result is negative is the specificity of the test. These quantities can be defined as shown below.

$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{Condition Positive}} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$P(TP) = \frac{TP}{TP + FN} \quad (5.1)$$

It can be noticed that the key to maximizing sensitivity is to minimize False Negatives. If the False Negatives = 0, then the Sensitivity is = 1. (Ideal Condition).

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{Condition Negative}} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

$$P(FP) = \frac{FP}{TN + FP} \quad (5.2)$$

Identical to the previous case, It can be noticed that the key to maximizing specificity is to minimize False Positives. If the False Positives = 0, then the Specificity is = 1. (Ideal Condition).

Therefore, the aim is to minimize False Negatives and False Positives. Doing so maximizes the Sensitivity and the Specificity.

Two other very important Bayesian probabilistic measures are the Positive Predictive Value (PPV) and the Negative Predictive Value (NPV). These can also be considered to be conditional probabilities. The probability of getting a true positive, given that the test result is positive is called the Positive Predictive Value (PPV). [14]

$$\text{Positive Predictive Value (PPV)} = \frac{\text{True Positives}}{\text{Predicted Positives}}$$

$$PPV = \frac{TP}{TP + FP} \quad (5.3)$$

The probability of getting a true negative, given that the test result is negative is called the Negative Predictive Value (NPV).

$$\text{Negative Predictive Value (NPV)} = \frac{\text{True Negatives}}{\text{Predicted Negatives}}$$

$$NPV = \frac{TN}{TN + FN} \quad (5.4)$$

5.3 Application of ROC

In our study, the dependent variable is the lifetime of the homology group generators. Hence it is required to come up with a threshold for the distance parameter for the Distance parameter. Three cut-off values are picked out on the basis of the output provided by the ROC MATLAB program made by Cardillo G. One lenient cut-off, one strict and one balanced cut-off.

The condition negatives and condition positives for the $\gamma = 0.36$ case corresponding to the datasets containing 100,200,303 and 400 points were fed into the ROC program. Three thresholds were picked and were benchmarked with the data corresponding to the Condition Negatives and Condition Positives for the data corresponding to $\gamma = 0.35, 0.37$ and 0.38 across datasets containing 100,200,303 and 400 points. The condition negatives and positives of the benchmarking data have been provided below.

Condition Negatives (Noise)									
0.000	0.001	0.002	0.004	0.005	0.013	0.018	0.027	0.052	0.077
0.000	0.001	0.002	0.004	0.006	0.013	0.019	0.027	0.055	0.081
0.001	0.001	0.002	0.004	0.007	0.013	0.020	0.028	0.056	0.086
0.001	0.001	0.003	0.004	0.007	0.013	0.021	0.029	0.056	0.093
0.001	0.001	0.003	0.004	0.010	0.013	0.022	0.031	0.057	0.108
0.001	0.001	0.003	0.004	0.010	0.014	0.022	0.033	0.059	0.113
0.001	0.002	0.003	0.005	0.011	0.014	0.023	0.036	0.064	0.138
0.001	0.002	0.003	0.005	0.011	0.014	0.023	0.037	0.065	0.145
0.001	0.002	0.003	0.005	0.012	0.014	0.024	0.042	0.071	0.160
0.001	0.002	0.003	0.005	0.012	0.015	0.024	0.045	0.071	0.161
0.001	0.002	0.004	0.005	0.013	0.017	0.024	0.047	0.073	0.167
0.001	0.002	0.004	0.005	0.013	0.017	0.027	0.048	0.073	0.168
0.001	0.002	0.004	0.005	0.013	0.018	0.027	0.048	0.073	0.173

Table 5.2: Condition Negatives of the Benchmarking Data

Condition Positives (Data)									
0.031	0.128	0.177	0.330	0.411	0.434	0.471	0.492	0.513	0.627
0.031	0.143	0.177	0.330	0.411	0.437	0.475	0.500	0.543	0.658
0.054	0.147	0.269	0.350	0.414	0.455	0.478	0.502	0.555	0.700
0.080	0.148	0.276	0.367	0.425	0.458	0.481	0.504	0.565	
0.083	0.149	0.330	0.392	0.427	0.466	0.483	0.509	0.568	

Table 5.3: Condition Positives of the Benchmarking Data

5.4 ROC Program

Upon inputting the condition positives of the $\gamma = 0.36$ case to the MATLAB program we get an output as shown in Table 5.4. The values of sensitivity and specificity are helpful in determining which cut-off to choose for the threshold value. The key is to keep both specificity and sensitivity high.

The program outputs a plot of "Sensitivity" ($P(TP)$) vs "1-Specificity" ($P(FP)$). The Area Under the Curve is 0.9968. This is almost equal to 1 which is ideal. This basically shows that the dataset is suitable for the use of ROC and is a measure of how "close" the condition positives and the condition negatives are placed in terms of the distance parameter.

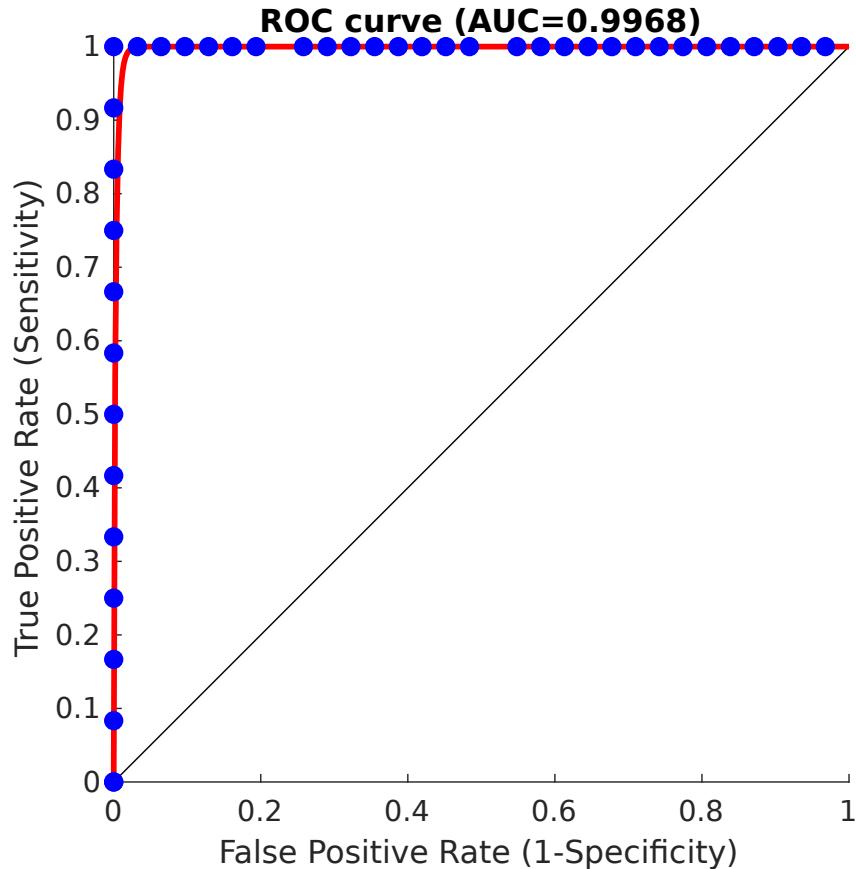


Figure 5.2: ROC Curve Output by MATLAB Program after

Cut-off	Sensitivity	Specificity	Efficiency
0.0025	1	0.29032	0.51163
0.003	1	0.35484	0.53488
0.0035	1	0.3871	0.55814
0.00375	1	0.41935	0.5814
0.005	1	0.45161	0.60465
0.0095	1	0.54839	0.67442
0.0085	1	0.51613	0.65116
0.0105	1	0.58065	0.69767
0.0175	1	0.6129	0.72093
0.02175	1	0.64516	0.74419
0.0255	1	0.67742	0.76744
0.0705	1	0.70968	0.7907
0.07125	1	0.74194	0.81395
0.076	1	0.80645	0.86047
0.0765	1	0.83871	0.88372
0.077	1	0.87097	0.90698
0.092	1	0.90323	0.93023
0.0995	1	0.93548	0.95349
0.1265	1	0.96774	0.97674
0.129	1	1	1
0.264	0.91667	1	0.97674
0.27225	0.83333	1	0.95349
0.3945	0.75	1	0.93023
0.3965	0.66667	1	0.90698
0.43125	0.58333	1	0.88372
0.438	0.5	1	0.86047
0.449	0.41667	1	0.83721
0.4535	0.33333	1	0.81395
0.6345	0.25	1	0.7907
0.695	0.16667	1	0.76744
0.77625	0.083333	1	0.74419

Table 5.4: Output of the ROC program

5.5 Choosing Cut-offs

The three cut-offs picked are 0.129, 0.264, 0.0705. These cut-off values are then tested against the case of $\gamma = 0.35, 0.37, 0.38$ from all the datasets namely, 100,200,303 and 400 points. The benchmark test proved to be successful with good values of sensitivity and specificity.

Cut-off = 0.0705			
Condition Negatives	129	True Negatives	114
		False Positives	15
Condition Positives	48	True Positives	45
		False Negatives	3
Sensitivity	0.938		
Specificity	0.883		

Table 5.5: Benchmaking data for threshold = 0.0705

Cut-off = 0.129			
Condition Negatives	129	True Negatives	123
		False Positives	6
Condition Positives	48	True Positives	42
		False Negatives	6
Sensitivity	0.875		
Specificity	0.953		

Table 5.6: Benchmaking data for threshold = 0.129

Cut-off = 0.264			
Condition Negatives	129	True Negatives	129
		False Positives	0
Condition Positives	48	True Positives	36
		False Negatives	12
Sensitivity	0.75		
Specificity	1.0		

Table 5.7: Benchmaking data for threshold = 0.129

5.5.1 Conclusion from ROC

The cut-off value of 0.129 is the most balanced value and it provides an excellent sensitivity of 0.875 and a specificity of 0.953. This ensures a balance between False Negatives (FN) and False Positives (FP). Thus it is possible to predict with reasonable accuracy the presence of loops in the system and hence predict the periodicity of the system.

CHAPTER 6

Conclusion

A framework of nonlinear dynamics, algebraic topology, statistical tools with a backbone of persistent homology has been presented in this thesis. This framework joins the dots between topological data analysis and dynamics and allows one to draw a simple yet powerful conclusion that the topology of the phase portrait can be used to predict the period of the system. It also shows how it is possible to predict the period of the system using just a few points on the phase space portrait.

Homology is a deep subject and this document merely scratches on the surface of the vast regime of established concepts. The concept of betti numbers though appearing as trivial at the first glance, yet it contains a complex and beautiful structure explained through algebraic topology. This concept when applied on a point cloud dataset using filtration and persistence, proves to be very useful to analyze the underlying topology of the data. The barcode plots produced after the filtration are intuitive and provide a pedagogically effective visualization of the topology of the dataset.

Physics revolves around dynamics, and it has been a powerful way to mathematically model systems to predict their behaviour. With the birth of powerful computers, various problems that were previously hard to solve due to the limitations of analytical solutions are now tackled using computational methods. Finding the period of oscillation of a system has held prime importance in solving various problems. Studying the topological

structures of the phase space plots provide great insight into the period of the system. The property that the number of loops in a system is equal to the period of the system has been linked with the fact that the rank of 1st Homology group is equal to the number of loops. Hence, a scheme to find the period of the system has been showcased.

The filtration obtained from the point cloud dataset produces both artefacts (noise) and data, to differentiate between these two a binary statistical test, namely ROC is employed. The noise and the data from a certain case is run through a computer program which outputs a options for threshold values. A few viable thresholds are chosen and then the chosen thresholds are benchmarked for specificity and sensitivity with the remaining cases to check the efficiency of the scheme.

Scope for Future Work

The future scope of this work is to employ this approach upon the chaotic regime of attractors and to characterize their dynamics using the information generated from the filtration of the point cloud dataset.

Another extremely interesting application is to characterize the bifurcations using Persistent Homology. And if possible to check for any pre-cursors the bifurcations by looking for correlations between Lyapunov exponent and the data generated from the filtration. Persistent Homology indeed does provide hope of a good future of solving various problems.

New statistical methods could be developed in the future to tackle the problem of interpreting lifetimes generated by the filtration.

Appendices

APPENDIX A

Programs

1.1 Introduction

This portion contains all the code used during the course of the project. MATLAB[15] was used all throughout. Here are a few programs that were used.

1.2 The Main Program

This program solves the duffing equation, extracts equispaced points, inputs them to the Javaplex plug-in which provides the barcode plots and also outputs the lifetimes generated from the filtration.

1.2.1 Solving the Duffing Equation

The program uses the unbuilt function `ode45` to solve the duffing equation and returns the solution in two arrays, one contains the time data and while the other contains the x and \dot{x} data. The phase space plot is generated by plotting the x and \dot{x} along the x and $y - axis$ respectively.

1.2.2 Extracting Equispaced Points

The algorithm skips points in the solution set array to generate a new array to produce 100, 200, 303 and 400 points. These points are generated in an equispaced manner. This forms the point cloud data.

1.2.3 Javaplex

The Javaplex[13] section of the program has been adapted from the official Javaplex tutorial.[16] It involves creating a stream of data using the point cloud data. This section outputs the intervals (lifetimes) and the barcode plots. A few parameters that can be changed are mentioned below along with the values that were used:

- Maximum Dimension of Homology Group Generators = 2
- Maximum filtration value = 1.5
- Number of Divisions = 2000

```

1 close all
2 clear
3 clc
4
5 load_javaplex; %initializes Javaplex
6
7 global delta alpha gamma beta omega
8
9 delta=0.3;
10 alpha=1;
11 beta=1;
12 omega=1.2;
13

```

```
14 % Use one of the lines given below, comment the other one
15 %GGAMM=[0.39]; % Single gamma value
16 %GGAMM=[0.35:0.01:0.38]; % Array of Gamma values
17
18 gm_size=size(GGAMM);
19
20 for i_g=1:gm_size(2)
21
22 j=100;
23
24 gamma=GGAMM(i_g);
25
26 [t x]=ode45(@duffing,0:2*pi/omega/1000:20000,[0 1]);
27 x_size=size(x);
28
29 low=10000;
30 high=20000;
31
32 figure,
33 subplot(1,2,1);
34
35 %Plots Phase space trajectory
36 plot(x(low:high,2),x(low:high,1),'b')
37 axis tight
38 axis equal
39 title(strcat(num2str(GGAMM(i_g)), 'phase space'));
40 ylabel({'$\dot{x}$'}, 'Interpreter', 'latex')
41 xlabel({'$x$'}, 'Interpreter', 'latex') % y-axis label
```

```
42
43 %Creates an array that skips j-number of points in the
    solution set
44 a(:,1)= x(low:j:high,1);
45 a(:,2)= x(low:j:high,2);
46
47
48 total = high-low;
49 count = floor((total)/j);      %Number of points
50 % disp(count);
51 % disp(total);
52
53 %PLots the points that have been used for sampling in
    Javaplex
54 subplot(1,2,2)
55 scatter(a(:,2),a(:,1),'.');
56 axis tight
57 axis equal
58 title(strcat((num2str(count)), ' of ', (num2str(total)), " ",
    'points'));
59 ylabel({'$\dot{x}$'},'Interpreter','latex')
60 xlabel({'$x$'},'Interpreter','latex') % y-axis label
61 print(strcat(num2str(GGAMM(i_g)), 'phase space', num2str(count
    )), '-dpng')
62 %section end
63
64
65 %Javaplex work Begins
66 import edu.stanford.math.plex4.*;
```

```
67
68 max_dimension = 2;
69 max_filtration_value = 1.0;
70 num_divisions = 2000;
71
72 % create the set of points
73
74 point_cloud = a;
75 size(point_cloud)
76
77 % create a Vietoris-Rips stream
78 stream = api.Plex4.createVietorisRipsStream(point_cloud,
    max_dimension, max_filtration_value, num_divisions);
79 num_simplices = stream.getSize()
80
81 % get persistence algorithm over Z/2Z
82 persistence = api.Plex4.getModularSimplicialAlgorithm(
    max_dimension, 2);
83
84 % compute the intervals
85 intervals = persistence.computeIntervals(stream);
86 disp(GGAMM(i_g));
87 disp(high);
88 disp(low);
89 disp(j);
90 disp(intervals);
91
92
93 val=gamma*10;
```

```

94
95 %Name for the title of the plot
96 name=strcat(num2str(GGAMM(i_g)*100), ' Barcode ', num2str(
97   count), ' ', 'Points');
98
99 % create the barcode plots
100 options.filename = name;
101 options.max_filtration_value = max_filtration_value;
102 options.max_dimension = max_dimension - 1;
103
104 plot_barcodes(intervals, options);
105
106 clear a
107
108 end

```

1.2.4 The Definition of the Duffing Oscillator

This code defines the ordinary differential equations of the duffing oscillator. The following code has been adapted from code published online at [Matlab Central Page](#) by Housam Binous.[17]

```

1
2 function xdot=duffing(t,x) % Function Name
3
4 global delta alpha gamma beta omega % variables
5
6 xdot(1)=-delta*x(1)+alpha*x(2)-beta*x(2)^3+gamma*cos(omega*t)
7   ; % Duffing Equation

```

```
7 xdot(2)=x(1); % Condition upon analytical solution  
8  
9 xdot=xdot';  
10  
11 end
```

REFERENCES

- [1] Jennifer Kloke. Methods and Applications of Topological Data Analysis. (May), 2010.
- [2] Gurjeet Singh, Facundo Mémoli, and Gunnar Carlsson. Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition. *Methods*, pages 91–100, 2007.
- [3] Khushboo Mittal and Shalabh Gupta. Topological characterization and early detection of bifurcations and chaos in complex systems using persistent homology. *Chaos*, 27(5):1–9, 2017.
- [4] Slobodan Maletić, Yi Zhao, and Milan Rajković. Persistent topological features of dynamical systems. *Chaos*, 26(5), 2016.
- [5] Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. Number January. 2006.
- [6] NJ Wildberger. An introduction to homology — Algebraic Topology — NJ Wildberger - YouTube.
- [7] Gunnar Carlsson. *Topology and data*, volume 46. 2009.
- [8] Prerna Nadathur. An Introduction to Homology. pages 1–14, 2007.
- [9] Gunnar Carlsson. Topological pattern recognition for point cloud data. *Acta Numerica*, 23:289–368, 2014.

- [10] ProboscideaRubber15. By ProboscideaRubber15 [CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>); from Wikimedia Commons, 2008.
- [11] Mahn-soo Choi. Group homomorphism, 2013.
- [12] Eric W. Weisstein. Duffing Differential Equation.
- [13] Andrew Tausz, Mikael Vejdemo-Johansson, and Henry Adams. Javaplex: A research software package for persistent (co)homology, 2014.
- [14] Andrew P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.
- [15] The Mathworks Inc. MATLAB, 2017.
- [16] Henry Adams and Andrew Tausz. Javaplex tutorial. pages 1–31, 2012.
- [17] Housam (National Institute of Applied Sciences Binous and TUNISIA) Technology, Tunis. No Title.