Topological Data Analysis

by
Madhuleka V Iyer

A dissertation submitted in partial fulfilment of the requirements for the degree of
B.Sc Honours in Mathematics
2020-2023

Supervised by: Divakaran D
Evaluated by: Divakaran D and Tulsi Srinivasan

Topological Data Analysis

Madhuleka V Iyer

Abstract

Over the course of this project, we studied Topology and explored an application of Topological Data Analysis. Topology is the study of properties that are preserved after deformations like stretching and twisting of objects. Topological Data Analysis is a method to analyse data by studying the shape of data. To do this, we convert the data into a simplicial complex. We then associate an algebraic invariant, Homology, to the simplicial complex. We study whether the generators of each Homology space persists or not. If it persists, we call this a feature of the data, else ignore it as noise.

We studied an application of Topological Data Analysis to human motion recognition. Through this we explored that Topological Data Analysis is a better way to classify motion recognition than Principal Component Analysis.

Dedication

To Amma, Appa, Namrata and Neela Patti. For supporting me and always believing in me.

Declaration

I hereby declare that the work in this thesis has been carried out by me, in the B.Sc. (Honours) Program, under the supervision of Dr. Divakaran D, and in the partial fulfillment of the requirements for the award of the degree of B. Sc (Honours) at the Azim Premji University, Bangalore. I further declare that this work has not been the basis for the award of any degree, diploma or any other title elsewhere.

Acknowledgements

I want to thank my honours guide and mentor, Divakaran, for being there every step of the way and being a pillar of support. I want to thank my mother and father for being the most supportive parents I could have asked for and always believing in me. I want to thank my sister and grandmother for their constant cheer. I also want to thank my friends for a joyous experience. It was an incredible journey with all these people by my side.

Contents

Chapter 1

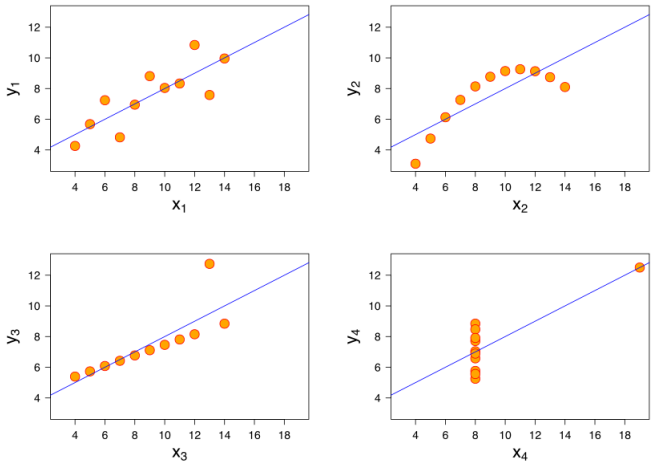
Introduction

Topological Data Analysis (TDA), the main theme of the project, is a method of analysing data by studying the shape of data using Topological methods. Given a data set X, we first convert the data set into a geometric object. Further, we assign an algebraic invariant to this geometric object. We study the properties of the geometric object through this invariant and learn something about the data.

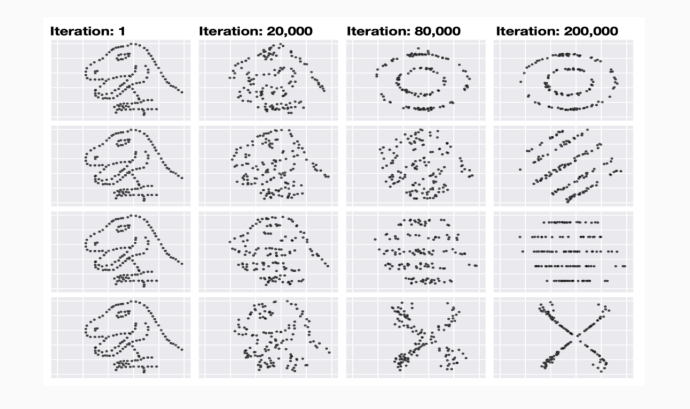
A natural question to ask here is, why should we study Topological Data Analysis when there are myriad other statistical techniques to analyse data? Many times, analysing data using summary statistics is insufficient. This is illuminated by the Anscombe's quartet and the Datasaurus.

The Anscombe's quartet has eleven data points with almost identical summary statistics. But the distributions look very different when plotted as a scatter plot, that is, they are qualitatively very different. The four graphs were devised by Francis Anscombe to show the importance of graphing in statistics.

The Datasaurus further shows us that the summary statistics can be deceiving. Certain data points are plotted as a scatter plot to take the shape of a dinosaur. The data points are then perturbed and moved towards a different shape. This is done while keeping the summary statistics intact up to two decimal points. After certain perturbations, the scatter plot looks like concentric circles, diagonal lines, horizontal lines and a cross. Since we ensured that the



summary statistics remain the same, these plots are quantitatively similar. However, upon plotting they look very different.



Though data sets are quantitatively similar, they can be qualitatively poles apart. So, we want to study the shape of data to distinguish between such data sets. We have claimed that summary statistics are insufficient. What about other statistical methods to analyse data?

The datasaurus does not lie on any curve. If the data points lie in a higher dimension, it is unlikely that the points would lie on a curve. Further, in Chapter 5, we will see that

even methods like Principal Component Analysis (PCA) are not efficient enough to classify data points effectively. We explore in Chapter 5 whether Topological Data Analysis is a better classifier of human motion recognition than Principal Component Analysis and learned it indeed is.

When we say that we want to study the shape of data by assigning geometric objects to it, what are these objects that we assign to the data? We associate a simplicial complex 2 to the data set and study the properties of this simplicial complex.

The aim of the project is to study the shape of data and explore an application of TDA. Before we can study the shape of data using topology, we need to know how to convert data into geometric objects, and how to assign an algebraic invariant to it. For this, we study homology 3 and persistence homology 4. For this, we read Vidit Nanda's Computational Algebraic Topology, Lecture notes Nanda n.d.

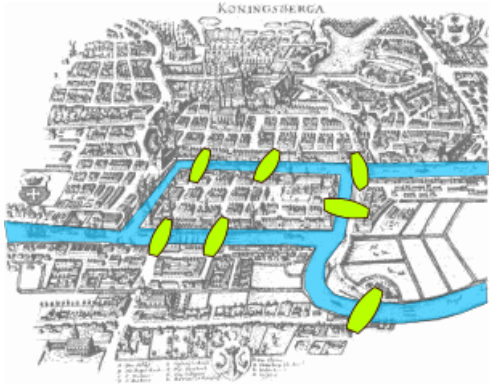
The thesis takes the reader through the basic concepts of Topology, equipping them with the knowledge required to perform TDA. Further, it elucidates the advantages of TDA over other methods that are traditionally used to perform similar data analysis. We explore this idea using an application of Topological Data Analysis to classify motion capture data.

Chapter 2

Simplicial Complex

2.1 Graphs

The famous Königsberg problem of seven bridges is one of the most important questions in the field of Discrete Mathematics. The city of Königsberg consisted of two islands and two mainland regions. Seven bridges connected the islands to each other and to the two mainland regions. A question arose as to whether it was possible to return to the starting point after traversing all seven bridges, travelling through each bridge precisely once. The seven bridges were constructed in the following way:



Leonhard Euler solved the problem in 1736. Upon coming across this puzzle, Euler under-

stood that the specific shapes or sizes of the four regions and the lengths of the seven bridges were irrelevant to the solution. So, he chose to represent the four regions as single points each and the bridges as line segments connecting the respective points. He proved that for such a traversal to exist, at least one region should have an even number of bridges connecting it to other regions. As all the regions in the given situation have an odd number of bridges connecting them to other regions, such a traversal is impossible. In this process, Euler invented Graph Theory and this led to the birth of Topology.

In general, the relationship between entities in a collection can be represented using graphs. Each entity is represented by a point and is called a vertex of the graph. If two entities are related, the relationship is expressed by a line segment connecting the two vertices and is called an edge of the graph.

Definition 1. *A graph G is an ordered pair (V, E) where V is a set of vertices and E is a set of two elements subsets of V.*

For example, consider a class with 6 students $\{A, B, C, D, E, F\}$. This forms a vertex set V. A line is drawn between two students if they have interacted with each other more than five times. A and B, A and C, B and C, D and E, and D and F have interacted with each other more than five times. Other pairs of students have interacted less than or equal to five times. This can be represented by the graph $G = (V, E)$ where $V = \{A, B, C, D, E, F\}$ and $E = \{\{A, B\}, \{A, C\}, \{B, C\}, \{D, E\}, \{D, F\}\}$.

If a further condition is added that A, B and C mutually interact with each other, this relationship can no longer be represented by graphs. This is because graphs only contain single-element sets (vertices) and double-element sets (edges) while a three-element set is required to represent this relationship. Therefore, graphs become an inadequate form of representation to model such relationships.

To study relationships in higher dimensions we require structures that include sets of higher cardinality. This motivates the study of simplicial complexes.

2.2 Simplicial Complex

Definition 2. *Given a set V, a simplicial complex K on V is a collection of non-empty subsets of the set V satisfying the following conditions:*

- *Every set containing only one vertex belongs to K*
- *If τ is in K and σ is a non-empty subset of τ, $\sigma \in K$.*

Example 2.2.1. *Let vertex set be $V = \{v_1, v_2, v_3, v_4\}$. Consider the set*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\}.$$

We must now check that K is indeed a simplicial complex. The set V has four elements. The simplicial complex K contains four singleton sets containing four vertices each. Hence the first condition is satisfied. Singleton sets do not have any non-empty subsets. Hence, the second condition is satisfied. Therefore, K is indeed a simplicial complex.

Example 2.2.2. *Going back to the example for graphs, the vertex set is $\{A, B, C, D, E, F\}$. Let us represent the graph G as*

$$K = \{\{A\}, \{B\}, \{C\}, \{D\}, \{E\}, \{F\}, \{A, B\}, \{A, C\}, \{B, C\}, \{D, E\}, \{D, F\}\}$$

where the singleton sets denote vertices and the two-element sets denote edges. We must now check that K is indeed a simplicial complex. The set V has six elements. The simplicial complex K contains six singleton sets containing six vertices each. Hence, the first condition is satisfied. Since K only has singleton sets and two-element sets, we need to check the second condition for these two kinds of sets. Singleton sets do not have any non-empty subsets. Two element sets can only have singleton subsets and since every possible singleton set is present in K, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Therefore, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Hence, the second condition is satisfied. Therefore, K is indeed a simplicial complex.

Example 2.2.3. *More generally, given a graph $G = (V, E)$, we can construct a set K containing n singleton sets and m two element sets where $|V| = n$ and $|E| = m$. Since the set contains n*

singleton sets, every vertex in the vertex set V is in K. Hence, the condition is satisfied. Since K only has singleton and two-element sets, we need to check the second condition for these two sets. Singleton sets do not have any non-empty subsets. Two element sets can only have singleton subsets and since every possible singleton set is present in K, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Hence, the second condition is satisfied. Therefore, K is indeed an analogous simplicial complex of the graph G.

Example 2.2.4. *Let vertex set be $V = \{v_1, v_2, v_3, v_4, v_5\}$. Consider the set*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}, \{v_2, v_4\}, \{v_4, v_5\}, \{v_1, v_2, v_3\}\}.$$

We must now check that K is indeed a simplicial complex. The set V has five elements. The simplicial complex K contains five singleton sets containing five vertices each. Hence the first condition is satisfied. Since K only has singleton sets, two-element sets and three-element sets, we need to check the second condition for these three kinds of sets. Singleton sets do not have any non-empty subsets. Two element sets can only have singleton subsets and since every possible singleton set is present in K, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Therefore, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. The only three elements set present in K is $\{v_1, v_2, v_3\}$. If this set is τ, then σ is any of the two-element or one-element subsets of the given set. The two element subsets include $\{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}$ and these are all present in K. Further, we have already confirmed that all possible singleton sets are present in K. Therefore, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Hence, the second condition is satisfied. Therefore, K is indeed a simplicial complex.

Example 2.2.5. *Let vertex set be $V = \{v_1, v_2, v_3, v_4\}$. Consider the set*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}$$

$$\{v_2, v_3\}, \{v_2, v_4\}, \{v_3, v_4\}, \{v_1, v_2, v_3\}, \{v_1, v_2, v_4\}, \{v_1, v_3, v_4\}, \{v_2, v_3, v_4\}.$$

We must now check that K is indeed a simplicial complex. The set V has four elements. The

simplicial complex K contains four singleton sets containing four vertices each. Hence the first condition is satisfied. Since K only has singleton sets, two-element sets and three-element sets, we need to check the second condition for these three kinds of sets. Singleton sets do not have any non-empty subsets. Two element sets can only have singleton subsets and since every possible singleton set is present in K, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Therefore, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. The three element sets present in K include $\{v_1, v_2, v_3\}, \{v_1, v_2, v_4\}, \{v_1, v_3, v_4\}, \{v_2, v_3, v_4\}$. If each of these three element sets is τ, then σ is any two-element or one-element subsets of the given set. The two element subsets of these three elements sets include $\{v_1, v_2\}, \{v_2, v_3\}, \{v_2, v_4\}, \{v_1, v_3\}, \{v_3, v_4\}, \{v_1, v_4\}$ and these are all present in K. Further, we have already confirmed that all possible singleton sets are present in K. Therefore, for all $\tau \in K$, if $\sigma \subset \tau$, $\sigma \in K$ as well. Hence, the second condition is satisfied. Therefore, K is indeed a simplicial complex.

Non-Example 2.2.1. *Let vertex set be $V = \{v_1, v_2, v_3, v_4, v_5\}$*
Consider the set

$$S = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}, \{v_3, v_4\}\}$$

.

We must now check if this set is a simplicial complex and we claim that it is not. The vertex set contains 5 vertices. For the first condition to be satisfied, all five vertices need to be present in S. But the vertex $\{v_5\}$ is not in S.

Hence, S is not a simplicial complex.

Non-Example 2.2.2. *Let vertex set be $V = \{v_1, v_2, v_3\}$.*

Consider the set

$$S = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_2, v_3\}\}$$

.

We must now check if this set is a simplicial complex and we claim that it is not. This set

contains the element $\{v_1, v_2, v_3\}$. *For the second condition of simplicial complexes to be satisfied, the edges* $\{v_1, v_2\}, \{v_2, v_3\} and \{v_1, v_3\}$ *need to be in* S. *The edge* $\{v_1, v_3\}$ *is not present in* S.

Hence, S *is not a simplicial complex.*

2.3 Simplex

Graphs are sets made up of two components, vertices and edges. We generalised in Example 2.2.3 that all graphs have analogous simplicial complexes where vertices corresponded to singleton sets and edges corresponded to two-element sets. Further, simplicial complexes contain higher cardinality sets as well. Each of these elements or sets of the simplicial complex are constituents that make up the simplicial complex. Breaking it back down to basic components, these sets or elements of the simplicial complex are called simplices.

Definition 3. *Any non-empty subset* σ *of* V *that lies in the simplicial complex* K *is called a simplex.*

Example 2.3.1. *Taking the first example from the last section, let the vertex set be* $V = \{v_1, v_2, v_3, v_4\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\}.$$

The subsets of V *that are also present in* K *include the sets* $\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}$. *Hence, these are all the simplices of* K.

Example 2.3.2. *Let vertex set be* $V = \{v_1, v_2, v_3, v_4\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_1, v_2, v_3\}\}.$$

The subsets of V *that are also present in* K *include the sets* $\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}$, $\{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_1, v_2, v_3\}$. *Hence, these are all the simplices of* K. *Consider the sets* $\{v_2, v_4\}, \{v_3, v_4\}, \{v_1, v_2, v_4\}$. *These are subsets of* V *but are not simplices since they are not present in* K.

Example 2.3.3. *Let vertex set* $V = \{v_1, v_2, v_3, v_4\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_1, v_3\}\}.$$

The subsets of V *that are also present in* K *include the sets* $\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_1, v_3\}\}$. *Hence, these are all the simplices of* K. *Consider the sets* $\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_2, v_3\}$. *These are subsets of* V *but are not simplices since they are not present in* K.

In the above examples, there are one-element, two-element and three-element subsets of V that are present in K. These are all simplices of K. Naturally, we would want to differentiate between the simplices of different cardinalities and hence define the dimension of simplices.

Definition 4. *The dimension of* σ *is* $|\sigma| - 1$, *that is, one less than the cardinality of* σ.

One element subsets of V have a dimension of zero, two element subsets of V have a dimension of one and so on. The dimension of the simplicial complex K is equal to the max $\{\dim \sigma | \sigma \in K\}$. We denote the set of all i-dimensional simplices in K as K_i. More specifically, the set of all single element sets or the set of all vertices is denoted by K_0.

Example 2.3.4. *Let vertex set* $V = \{v_0, v_1, v_2\}$. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_0, v_1\}, \{v_1, v_2\}, \{v_0, v_2\}, \{v_0, v_1, v_2\}\}.$$

The simplices of K *are* $\{v_0\}, \{v_1\}, \{v_2\}, \{v_0, v_1\}, \{v_1, v_2\}, \{v_0, v_2\}, \{v_0, v_1, v_2\}$. *The simplices* $\{v_0\}, \{v_1\}, \{v_2\}$ *have a dimension of zero,* $\{v_0, v_1\}, \{v_1, v_2\}, \{v_0, v_2\}$ *have a dimension of one and* $\{v_0, v_1, v_2\}$ *has a dimension of two.*

Some of the aforementioned n-dimensional simplices are subsets of higher dimensional simplices. When this happens, we say that the n-dimensional simplices are faces of the higher dimensional simplices.

Definition 5. *Given two simplices* σ *and* τ, σ *is a face of* τ $(\sigma \leq \tau)$ *when every vertex of* σ *is also a vertex of* τ.

Example 2.3.5. *Let vertex set* $V = \{v_1, v_2, v_3\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}, \{v_1, v_2, v_3\}\}.$$

We claim that the zero-dimensional simplices, $\{v_1\}$ *and* $\{v_2\}$ *are the faces of the two dimensional simplex* $\{v_1, v_2\}$. *Further, we also claim that the two-dimensional simplices* $\{v_1, v_2\}, \{v_2, v_3\}$ *and* $\{v_1, v_3\}$ *are faces of the three-dimensional simplex* $\{v_1, v_2, v_3\}$. *This is true since every vertex in the former simplices is also present in the latter simplex.*

Example 2.3.6. *Let vertex set* $V = \{v_1, v_2, v_3\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}\}.$$

Every simplex is of dimension zero and has only one element. Therefore, none of the simplices has a face other than themselves.

We defined dimension to facilitate the differentiation of simplices with different cardinalities. Faces are also of different cardinalities and to differentiate them, we define a term called the co-dimension.

Definition 6. *Let* σ *and* τ *be a pair of simplices of the simplicial complex* K *where* σ *is a face of* τ. *The co-dimension of* σ *as a face of* τ *is* $\dim \tau - \dim \sigma$.

The dimensions of the faces of a simplex are always less than or equal to the dimension of the simplex.

Example 2.3.7. *Taking the first example from the previous section, Let vertex set* $V = \{v_1, v_2, v_3\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}, \{v_1, v_2, v_3\}\}.$$

The faces of the simplex $\{v_1, v_2\}$ *are* $\{v_1\}$ *and* $\{v_2\}$. *Here,* τ *is the one-dimensional simplex and* σ *is a zero-dimensional face of the simplex. The co-dimension of the face is* $\dim \tau - \dim \sigma = 1 - 0 = 1$.

A face of the simplex $\{v_1, v_2, v_3\}$ *is* $\{v_1, v_2\}$*. Here,* τ *is the two-dimensional simplex and* σ *is the one-dimensional face of the simplex. The co-dimension of the face is* $dim\ \tau - dim\ \sigma = 2-1 = 1$*. Further,* $\{v_1\}$ *is also a face of the simplex* $\{v_1, v_2, v_3\}$*. Here,* τ *is the two-dimensional simplex and* σ *is the zero-dimensional face of the simplex. The co-dimension of the face is* $dim\ \tau - dim\ \sigma = 2 - 0 = 2$*.*

From the examples, it is clear that sets of the same cardinality can have different co-dimensions when considered as a face of simplices of different dimensions.

2.4 Directed Graphs and Oriented Simplicial Complexes

We motivated the study of simplicial complexes as higher dimensional analogues of graphs. Further, we defined the edge set in graphs as a set of two-element subsets of V. Sets are generally unordered. So, the order of the elements in each element of E is unimportant in graphs. We define directed graphs that are a special case of graphs, where the elements in every element of E has an ordering. The set E will no longer remain a set of subsets of V. The set E will now be a set of ordered pairs where the elements of the ordered pair are from V.

Definition 7. *A directed graph is an ordered pair* (V, E) *where* V *is a set of vertices and* E *is a subset of* $V \times V$*.*

Consider the set $V = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Let us take an example where an edge is drawn from from vertex A to vertex B if A is divides B. In this case, the set E will be $\{(1,2), (1,3), (1,4), (1,5), (1,6), (1,7), (1,8), (1,9), (1,10), (2,4), (2,6), (2,8), (2,10), (3,6), (3,9), (4,8), (5,10)\}$

Here, the order of the elements in the tuple matters. While there is an edge $(2,4)$, there is no edge $(4,2)$.

The first element in the pair is assigned a negative sign and is called the source of the edge. The second element in the pair is assigned a positive sign and is called the sink of the edge. An edge can have only one source and one sink. Not all vertices need to be a source or a sink.

Oriented simplicial complexes are a higher dimensional analogue of directed graphs.

We write simplices as subsets of vertices. But do we write the vertices in any particular order? The vertices are all written with sub-scripts. But do these sub-scripts signify anything or are they just for differentiation?

If the vertices are written in a particular order, then such a simplex is called an oriented simplex. But the order need not be the same as the order of the sub-scripts. We defined a function $o : K_0 \to \mathbb{N}$ which assigns a unique natural number to each vertex. The natural order on the set of natural numbers induces an order of the set of vertices. Since we are concerned with the order of the vertices and not the number that is assigned to the vertex itself, we only assign the first m natural numbers where m is the cardinality of the set of all zero dimensional simplices.

The simplex σ is hereafter written as an ordered subset of the set of all zero-dimensional simplices. For the sake of simplicity, $o(v_0) < o(v_1) < \ldots o(v_k)$. Simplicial complexes consisting of oriented simplices form an oriented simplicial complex.

We have been talking about simplicial complexes, simplices and their characteristics. Developing on these ideas, we would like to visualise these abstract structures. To be able to pictorially represent these structures, we shall define maps from the set of all zero-dimensional simplices or vertices in K to $\mathbb{R}^n$.

2.5 Geometric Realisation

To geometrically realise the abstract simplicial complexes, we equate zero-dimensional simplices to vertices, one-dimensional simplices to edges, two-dimensional simplices to filled triangles and so on.

While we want to geometrically realise all simplices, we define a map only on the zero-dimensional simplices. This map takes vertices to points in $\mathbb{R}^n$. The higher dimensional geometric simplices are the convex hull of the constituent vertices.

Definition 8. *Given a set of vertices* $\{x_0, x_1, \ldots, x_n\}$*, the set* $\{t_1x_1 + t_2x_2 + \ldots t_nx_n \mid t_1 + t_2 + \cdots + t_n = 1$ *and* $t_1, t_2, \ldots t_n \geq 0\}$ *is the convex hull of these vertices.*

The function that maps the vertices to points in $\mathbb{R}^n$ further needs to follow a few conditions. For example, intuitively, for an object with more than 2 vertices to be 2 dimensional, the vertices should not line on the same line. This means that a 2-dimensional simplex cannot be mapped onto a line in R^2.

Any three vertices $\{x_0, x_1, x_2\}$ do not lie on the same line iff $(x_1 - x_0)$ and $(x_2 - x_0)$ are linearly independent. In general, $n + 1$ points do not lie on an $n - 1$ - dimensional subspace iff the vectors $(x_1 - x_0), (x_2 - x_0), \ldots, (x_n - x_0)$ are linearly independent. We term vertices satisfying such a linear independence as being affinely independent.

Definition 9. *The points $\{x_0, x_1, x_2, \ldots, x_n\}$ are said to be affinely independent if the set of vectors $\{(x_1 - x_0), (x_2 - x_0), \ldots, (x_n - x_0)\}$ is linearly independent.*

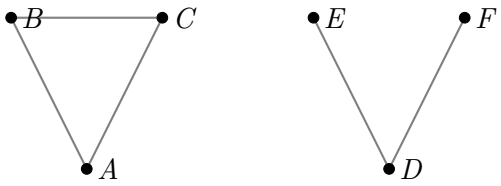
Further, we would not want two points to be mapped to the same element in $\mathbb{R}^n$. This is because such a mapping would lead to a reduction in dimension. Therefore, the function from the set of all zero-dimensional simplices in K to $\mathbb{R}^n$ must be injective. Let us now define the function.

Definition 10. *Take a function $\varphi : K_0 \to \mathbb{R}^n$. This function maps the zero-dimensional simplices of K onto a point in $\mathbb{R}^n$. The geometric realisation $|\sigma|_\varphi$ refers to the mapping of the simplex σ onto $\mathbb{R}^n$ with respect to the function φ. Here, $\sigma = \{v_1, v_2, \ldots v_k\}$ and $|\sigma|_\varphi$ is the geometric simplex of spanned by $\{\varphi(v_0), \varphi(v_1), \varphi(v_2), \ldots, \varphi(v_k)\}$.*

When the function φ is injective and $\varphi(K_0)$ are affinely independent, we call the function φ an affine embedding of K in $\mathbb{R}^n$. Since K is a union of its constituent simplices, the union of $|\sigma|_\varphi$ is the geometric realisation of K with respect to the function φ.

To understand the examples from Graphs 2.1 and Simplicial Complex 2.2 sections pictorially, we take the same examples and define the function $\varphi : K_0 \to \mathbb{R}^n$ on them.

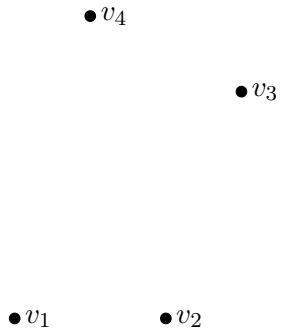
Example 2.5.1. *Let vertex set $V = \{A, B, C, D, E, F\}$ and edge set $E = \{(A, B), (A, C), (B, C), (D, E), (D, F)\}$. The function $\varphi : K_0 \to \mathbb{R}^n$ is defined as $\varphi(A) = (2, 3), \varphi(B) = (1, 5), \varphi(C) = (3, 5), \varphi(D) = (6, 3), \varphi(E) = (5, 5)$, and $\varphi(F) = (7, 5)$. Upon mapping, we get*



Example 2.5.2. *Let vertex set be* $V = \{v_1, v_2, v_3, v_4\}$. *Consider the set*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\}.$$

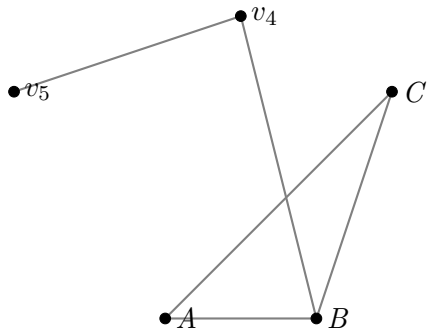
The function $\varphi : K_0 \to \mathbb{R}^n$ *is defined as* $\varphi(v_1) = (0,0), \varphi(v_2) = (2,0), \varphi(v_3) = (3,3)$, *and* $\varphi(v_4) = (1,4)$. *Upon mapping, we get*



Example 2.5.3. *Let vertex set be* $V = \{v_1, v_2, v_3, v_4, v_5\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}, \{v_2, v_4\}, \{v_4, v_5\}, \{v_1, v_2, v_3\}\}.$$

The function $\varphi : K_0 \to \mathbb{R}^n$ *is defined as* $\varphi(v_1) = (0,0), \varphi(v_2) = (2,0), \varphi(v_3) = (3,3), \varphi(v_4) = (1,4)$, *and* $\varphi(v_5) = (-2,3)$. *Upon mapping, we get*



Non-Example 2.5.1. *Let vertex set be* $V = \{v_0, v_1, v_2\}$. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_0, v_1\}, \{v_1, v_2\}, \{v_0, v_2\}, \{v_0, v_1, v_2\}\}.$$

The function $\varphi : K_0 \to \mathbb{R}^2$ *is defined as* $\varphi(v_0) = (0,0), \varphi(v_1) = (0,0)$ *and* $\varphi(v_2) = (0,0)$.

Here, all elements are mapped to the origin. This function is not an affine embedding as φ *is not an injective function. Therefore, it is not a geometric realisation of the simplicial complex.*

Non-Example 2.5.2. *Let vertex set be* $V = \{v_0, v_1, v_2\}$. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_0, v_1\}, \{v_1, v_2\}, \{v_0, v_2\}, \{v_0, v_1, v_2\}\}.$$

The function $\varphi : K_0 \to \mathbb{R}^2$ *is defined as* $\varphi(v_0) = (0,0), \varphi(v_1) = (1,0),$ *and* $\varphi(v_2) = (2,0)$
This function is not an affine embedding as the images of φ *are not affinely independent as the image is not affinely independent. Therefore, it is not a geometric realisation of the simplicial complex.*

As with any algebraic structure, we want to understand what happens when we take a certain part of the structure and work with the same. It is natural that for this sub-structure to retain its properties, it needs to maintain its characteristics. In the next section, we explore sub-complexes.

2.6 Sub-Complex

Definition 11. *A subset* L *of* K *is a sub-complex of* K *if it is also a simplicial complex.*

Example 2.6.1. *Let vertex set* $V = \{v_1, v_2, v_3\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}\}.$$

A subset L *of* K *is taken where* $L = \{\{v_1\}\}$. *The subset* L *is clearly a simplicial complex and is therefore a sub-complex of* K.

Example 2.6.2. *Let vertex set* $V = \{v_1, v_2, v_3\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}, \{v_1, v_2, v_3\}\}.$$

A subset L *of* K *is taken where* $L = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}\}$. *The subset* L *is clearly a simplicial complex and is therefore a sub-complex of* K.

Consider another subset M *of* K *where* $M = \{\{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}\}$. *This is clearly not a simplicial complex since* $\{v_1, v_2\}$ *is in* M *while* $\{v_1\}, \{v_2\}$ *are not in* M *and is therefore not a sub-complex of* K.

Example 2.6.3. *Let vertex set* $V = \{v_1, v_2, v_3\}$. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}, \{v_1, v_2, v_3\}\}.$$

A subset L *of* K *is taken where* $L = \{\{v_1\}, \{v_2\}, \{v_3\}\}$. *The subset* L *is clearly a simplicial complex and is therefore a sub-complex of* K.

Consider another subset M *of* K *where* $M = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}\}$. *This is clearly not a sub-complex of* K *since none of the vertices is included in* M *and is therefore not a sub-complex of* K.

2.7 Closure

The examples of the subsets M of K in the previous section were not sub-complexes of K. But all these subsets can be extended into sub-complexes by adding some elements to these subsets. Such an extended subset is called a closure of the original subset.

Definition 12. *Let* K' *be a subset of the simplicial complex* K. L *is a closure of* K' *if* L *is the smallest sub-complex containing* K'.

Example 2.7.1. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}\}.$$

Consider the subset $K' = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}\}$. *Here, the vertices are missing and therefore,* K' *is not a sub-complex. Upon adding the vertices to the set* K', *it will become a simplicial complex and thereby a sub-complex of* K. *Therefore, the closure of* K' *is* $L = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}\}$.

Example 2.7.2. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}\}.$$

Let us consider the set $K' = \{\{v_1\}, \{v_2\}\}$. *This set is a simplicial complex and is therefore a sub-complex of* K'. *Therefore, its closure is the set* K' *itself, that is* $L = K'$.

Theorem 1. K' *is the closure of itself iff* K' *is a sub-complex.*

Proof. The proof of this theorem is twofold. We must prove that if K' is the closure of itself then K' is a sub-complex and that if K' is a sub-complex then K' is the closure of itself.

Let us first prove the first statement. Let us assume that K' is the closure of K. This means that K' is the smallest sub-complex containing itself. This means that K' is a sub-complex.

Now let us prove the second statement. Let us assume that K' is a sub-complex. Let us take any L that is any sub-complex of K and contains K'. This means that $K' \subseteq L$. This implies that K' is smaller than or equal to L. Since K' is a sub-complex of K and for all L containing K', K' is smaller than L, K' is the smallest sub-complex containing K'. □

This leads us to the question of whether the 'smallest' sub-complex exists. For example, if we take two sub-complexes L and M containing K' such that $L \not\subset M$ and $M \not\subset L$, then we cannot compare the two sets. We claim that we can find a sub-complex smaller than both L and M containing K', namely, $L \cap M$. The next step in this process is to prove that $L \cap M$ is a simplicial complex.

Claim 1. $L \cap M$ *is a simplicial complex if* L *and* M *are simplicial complexes.*

Proof. To prove that $L \cap M$ is a sub-complex, we must prove that every singleton subset of V is contained in $L \cap M$ and that if $\tau \in L \cap M$ and $\sigma \subset \tau$, then $\sigma \in L \cap M$.

Proving the first part, let us first show that every singleton subset of V is contained in $L \cap M$. We know that L and M are simplicial complexes. Hence, every vertex is contained in L and is contained in M. Hence, every vertex is contained in $L \cap M$.

To prove the next part, let us show that if $\tau \in L \cap M$ and $\sigma \subset \tau$, then $\sigma \in L \cap M$. If $\tau \in L \cap M$, then $\tau \in L$ and $\tau \in M$. Let us first consider the case where τ in L. This implies that if $\sigma \subset \tau$ then $\sigma \in L$ as L is a simplicial complex. Next Similarly, if τ in M, and $\sigma \subset \tau$ then $\sigma \in M$ as M is a sub-complex. Since, $\sigma \in L$ and $\sigma \in M$, $\sigma \in L \cap M$. Therefore, if $\tau \in L \cap M$ and $\sigma \subset \tau$, then $\sigma \in L \cap M$. □

The next question that we would like to answer is if we can generalise this result to say that $\bigcap_{L \in \mathcal{L}} L$ is a sub-complex if for all $L \in \mathcal{L}$, L is a sub-complex.

Claim 2. *$\bigcap_{L \in \mathcal{L}} L$ is a simplicial complex if for all $L \in \mathcal{L}$, L is a simplicial complex.*

Proof. To prove that $\bigcap_{L \in \mathcal{L}} L$ is a sub-complex, we must prove that every singleton subset of V is contained in $\bigcap_{L \in \mathcal{L}} L$ and that if $\tau \in \bigcap_{L \in \mathcal{L}} L$ and $\sigma \subset \tau$, then $\sigma \in \bigcap_{L \in \mathcal{L}} L$.

Proving the first part, we need to show that every singleton subset of V is contained in $\bigcap_{L \in \mathcal{L}} L$. We know that for all $L \in \mathcal{L}$ $\bigcap_{L \in \mathcal{L}} L$ is a simplicial complex. Hence, every vertex is contained in L for all $L \in \mathcal{L}$. Hence, every vertex is contained in $\bigcap_{L \in \mathcal{L}} L$. Proving the second part, we need to show that if $\tau \in \bigcap_{L \in \mathcal{L}} L$ and $\sigma \subset \tau$, then $\sigma \in \bigcap_{L \in \mathcal{L}} L$. If $\tau \in \bigcap_{L \in \mathcal{L}} L$, then $\tau \in L$ for all $L \in \mathcal{L}$. If $\tau \in L$ for all $L \in \mathcal{L}$ and if $\sigma \subset \tau$ then $\sigma \in L$ as L is a simplicial complex. Since, for all $L \in \mathcal{L}$, $\sigma \in L$, $\sigma \in \bigcap_{L \in \mathcal{L}} L$.

Hence, $\bigcap_{L \in \mathcal{L}} L$ is a sub-complex. $\mathcal{L} = \{K' \subset L | \text{L is a sub-complex}\}$. □

Therefore, given a subset K' of K such that sub-complexes L_i contains K' for $i \in \mathbb{N}$, then the closure of K' is $\bigcap_{L \in \mathcal{L}} L$.

Chapter 3

Homology

3.1 Euler Characteristic

We began the previous chapter with the problem of the seven bridges that Leonhard Euler resolved using a novel method now known as Graph Theory. We begin this chapter with another of Euler's many contributions to the field of Mathematics.

In school, we have encountered the Euler number for convex polyhedra. If V is the number of vertices, E is the number of edges and F is the number of faces in the convex polyhedron, then $V - E + F$ is the constant 2. The Euler number is a characteristic or property of convex polyhedra that remains unchanged. Such characteristics or properties are called an invariant of the object. In general, the Euler characteristic defined for different geometric objects is an invariant. We are interested in the Euler characteristic of Simplicial Complexes.

Definition 13. *The Euler characteristic of a simplicial complex K is the integer $\chi(K) \in \mathbb{Z}$ given by the alternating sum of cardinalities*

$$\chi(K) = \sum_{i=0}^{dimK} (-1)^i \#K_i$$

where $\#K_i$ is the number of i dimensional simplices in K.

Example 3.1.1. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}\}.$$

The number of 0-dimensional simplices is 3 and there are no other higher-dimensional simplices. So,

$$\chi(K) = \sum_{i=0}^{1} (-1)^i \# K_i.$$

This gives us, $(-1)^0 \times 4 = 4$. *Therefore, the Euler characteristic of* K *is 4. In the case where the simplicial complex only consists of 0-dimensional simplices, the Euler characteristic is just the number of simplices in* K.

Example 3.1.2. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\{v_0, v_1\}\}.$$

The number of 0-dimensional simplices is 5 and the number of 1-dimensional simplices is 1. There are no other higher dimensional simplices. So,

$$\chi(K) = \sum_{i=0}^{1} (-1)^i \# K_i.$$

This gives us, $(-1)^0 \times 5 + (-1)^1 \times 1 = 5 - 1 = 4$. *Therefore, the Euler characteristic of* K *is 4.*

The Euler characteristics of both examples are 4. But clearly, the two simplicial complexes are not the same. So, having the same Euler characteristic does not guarantee that the two simplicial complexes are equivalent in any way.

While having the same Euler characteristic does not guarantee that the two simplicial complexes are equivalent, if two simplicial complexes are equivalent, they will have the same Euler characteristic.

By assigning such invariants, we reduce the simplicial complexes to an algebraic value. In this chapter, we will be looking at a much stronger invariant called Homology. To define and

study homology, we need to understand chains and boundaries of simplicial complexes.

3.2 Chains

We have all encountered polynomials during our journey with Mathematics. Polynomials, when considered an algebraic object, are written as $a_0x^0+a_1x^1+\ldots a_nx^n$. Now consider the polynomial $x + x^2$. Here, the symbol $'+'$ does not indicate addition. There is no prescribed way to add x^2 to x. It is a way of indicating that there is a certain combination of the two elements. Such a sum is called the formal sum of the elements. A formal linear combination is the formal sum of basis elements multiplied by co-coefficients from the respective field.

A k-chain of a simplicial complex is a formal linear combination of the constituent k dimensional simplices with co-coefficients from a field $\mathbb{F}$. These chains are elements of the set C_k. The set C_k is a vector space over the field $\mathbb{F}$ where the field can be $\mathbb{R}, \mathbb{Z}/2\mathbb{Z}$ and so on. It is also possible to take the co-coefficients from a ring and a popular choice for this is $\mathbb{Z}$. In this situation, C_k is not a vector space but is a module. For the purposes of this project, we continue to treat C_k as a vector space over some field $\mathbb{F}$. The vector space C_k is generated by the set of all k-simplices in the simplicial complex K.

Definition 14. *For each dimension, $k \geq 0$, the k-th chain group of K is the vector space $C_k(K)$ over $\mathbb{F}$ generated by treating the k-simplices of K as a basis.*

Example 3.2.1. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}\}.$$

The only simplices in K are 0-dimensional. So, the only chains in K are 0-chains and the set $C_0 =< v_0, v_1, v_2 >$. The simplicial complex does not contain any 1-chains. So, C_1 is a vector space that is generated by the null set. This means that C_1 will be a zero-dimensional vector spaces containing only the zero elements. Similarly, there are no higher chains in K either. So, they are all generated by the null set. The sets C_i for all $i \geq 1$ are zero-dimensional vector spaces, that is, they are singleton sets containing only the zero elements.

Example 3.2.2. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}, \{v_1, v_2, v_3\}\}.$$

All 0-dimensional simplices include $\{v_0\}, \{v_1\}, \{v_2\}$ *and all 1-dimensional simplices include* $\{v_1, v_2\}, \{v_1, v_3\}$. *The only simplices in* K *are of zero, one or two dimensions. So, the only chains in* K *are zero, one or two chains. The set* $C_0 =< v_0, v_1, v_2 >$, $C_1 =< v_1v_2, v_1v_3, v_2v_3 >$ *and* $C_2 =< v_0v_1v_2 >$.

The simplicial complex does not contain any 3-chains. So, C_3 *is a vector space that is generated by the null set. This means that* C_3 *will be a zero-dimensional vector spaces containing only the zero elements. Similarly, there are no higher chains in* K *either. So, they are all generated by the null set. The sets* C_i *for all* $i \geq 3$ *are zero-dimensional vector spaces, that is, they are singleton sets containing only the zero elements.*

Now that we have understood chains, the next thing to understand before defining the big invariant Homology, is boundaries.

3.3 Boundaries

Consider the chain $v_0v_1 + v_1v_2$. What will the boundary of this chain be? Pictorially, the chain will look as follows:

A reasonable guess for the boundary of the above chain would be that it is some combination of its extreme vertices, v_0 and v_2. Let us represent the boundary map by ∂. Considering the boundary of a 1 dimensional simplex to be some linear combination of its vertices, we feel that, the co-coefficients should not depend on the choice of the 1 dimensional simplex. So, $\partial(v_0v_1) = \alpha_1v_0 + \alpha_2v_1$ and $\partial(v_1v_2) = \alpha_1v_1 + \alpha_2v_2$. Now, let us look at the boundary of the chain $v_0v_1 + v_1v_2$. The boundary map must be a linear map since the set of all k dimensional chains

is a vector space. Considering linearity for now, we get that,

$$\partial(v_0v_1 + v_1v_2) = \partial(v_0v_1) + \partial(v_1v_2) = \alpha_1v_0 + \alpha_2v_1 + \alpha_1v_1 + \alpha_2v_2 = \alpha_1v_0 + (\alpha_1 + \alpha_2)v_1 + \alpha_2v_2.$$

We already established that, intuitively, the boundary of the above simplicial chain should involve v_0 and v_2 but not v_1. So, we would want $\alpha_1 + \alpha_2 = 0$ or $\alpha_2 = -\alpha_1$. Thus, $\partial(v_0v_1) = \alpha v_1 - \alpha v_0$ for some α. The most natural choice for α is 1. So, we may define $\partial(v_0v_1) := v_1 - v_0$. This observation can be generalised to higher dimensional simplices as well and we may formalise this observation into an abstract definition of the boundary map.

The boundary of an n dimensional simplex, intuitively, is the constituent $n-1$ simplices. So, the boundary of k chains consists of $k-1$ chains. We have previously studied oriented simplices and we denote them by σ. Let us now define σ_{-i}.

Definition 15. *Let K be an oriented simplicial complex and let $\sigma = \{v_0, v_1, \dots v_k\}$ be an oriented k-simplex in K. For each i in $\{0 \dots k\}$, the i-th face of σ is the $k-1$ dimensional simplex*

$$\sigma_{-i} = \{v_0, v_1, \dots v_{i-1}, v_{i+1}, \dots v_k\}$$

obtained by removing the i-th vertex.

Now that we have defined σ_{-i}, let us define the boundary map of a k-chain.

Definition 16. *For each dimension $k \geq 0$, the k-th boundary operator of K is the $\mathbb{F}$- linear map $\partial_k^K : C_k(K) \to C_{k-1}(K)$ which sends each basis k-chain σ to the $(k-1)$ chain.*

$$\partial_k^K(\sigma) = \sum_{i=0}^{k} (-1)^i \sigma_{-i}$$

The boundary map is only defined on the basis of k-chains. We extend this map linearly to the other elements in the k chain vector space.

Example 3.3.1. *Consider the vertex set $\{v_0, v_1, v_2, v_3, v_4\}$. Consider the simplicial complex $\{\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_0, v_1\}, \{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}\}$. We have $C_0 =< v_0, v_1, v_2, v_3, v_4 >$,*

$C_1 =< v_0v_1, v_1v_2, v_2v_3, v_3v_4 >$. *These are ordered basis of* K. *We want to find the boundary of the chain* $v_0v_1 + v_1v_2 + v_2v_3 + v_3v_4$.

The boundary of the above chain will be $v_1 - v_0 + v_2 - v_1 + v_3 - v_2 + v_4 - v_3 = v_4 - v_0$.

Therefore, the boundary of the chain $v_0v_1 + v_1v_2 + v_2v_3 + v_3v_4$ *is* $v_4 - v_0$.

We can represent the example 3.3.1 in the form of matrices where the columns represent the edges and the rows represent the vertices. The vertex that is the sink of the edge takes the value of 1 and the vertex that is the source takes the value of -1. All other vertices take the value of 0. Each edge can have only one source and only one sink. So, each column has exactly one positive 1 and one negative 1. Every other element in the column is a 0.

Example 3.3.2. *The following matrix captures this information in a concise manner.*

$$\begin{bmatrix} & e_0 & e_1 & e_2 & e_3 \\ v_0 & -1 & 0 & 0 & 0 \\ v_1 & 1 & -1 & 0 & 0 \\ v_2 & 0 & 1 & -1 & 0 \\ v_3 & 0 & 0 & 1 & -1 \\ v_4 & 0 & 0 & 0 & 1 \end{bmatrix}$$

We have defined the boundary of k chains to be $k-1$ chains. We can find the boundaries of boundaries as well. While the function ∂_k represents the boundary map of k simplices, ∂_{k-1} represents the boundary map of $k-1$ simplices. We claim that the boundary of a boundary map is equivalent to 0. We prove this as follows.

Proposition 2. *For any oriented simplex* σ *of dimension* $k \geq 0$, *we have*

$$\partial_{k-1} \circ \partial_k(\sigma) \equiv 0.$$

Proof. We know that $\partial_k(\sigma) = \sum_{i=0}^{k} (-1)^i \sigma_{-i}$. We now need to find the boundary of $\partial_k(\sigma)$. We write this as

$$\partial_{k-1}(\sum_{i=0}^{k} (-1)^i \sigma_{-i}).$$

Expanding this, we get,

$$\sum_{i=0}^{k}(\sum_{j<i}(-1)^{j}(\sigma_{-ij}) + \sum_{j>i}(-1)^{j-1}(\sigma_{-ij})).$$

This can be written as,

$$\sum_{j<i}(-1)^{i+j}(\sigma_{-ij}) + \sum_{j>i}(-1)^{i+j-1}(\sigma_{-ij}).$$

The element σ_{-ij} is the same in both the summations. But, one of them has a co-efficient of $(-1)^{i+j}$ and the other has a co-efficient of $(-1)^{i+j-1}$. This means that one of them will have a co-efficient of 1 and the other will have a co-efficient of -1. Effectively, the two values get canceled out. So,

$$\sum_{j<i}(-1)^{i+j}(\sigma_{-ij}) + \sum_{j>i}(-1)^{i+j-1}(\sigma_{-ij}) = 0.$$

Hence, we have proven that $\partial_{k-1} \circ \partial_k(\sigma) \equiv 0$.

□

3.4 Homology

Homology is an invariant that is used to algebraically capture the essence of a simplicial complex. Previously, we had said that we need to study chains and boundaries to define and understand homology. This is because the homology of a simplicial complex is defined as the quotient vector space of the ker ∂_k by the Im ∂_{k+1}.

Definition 17. *For each dimension $k \geq 0$, the k-th homology group of a simplicial complex K is defined to be the quotient vector space*

$$H_k(K) = ker\ \partial_k / Im\ \partial k + 1$$

Both the ker ∂_k and the Im ∂_{k+1} are sub-spaces of C_k. The elements in the ker ∂_k are called k-cycles and the elements in the Im ∂_{k+1} are called k-boundaries.

Example 3.4.1. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\}.$$

The zero dimensional simplices in K include four elements, $\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}$ *and* $\{v_4\}$*. So, the vector space* $C_0 =< v_0, v_1, v_2, v_3 >$*. The kernel of* $\partial_0 = C_0$*. There are no 1-dimensional simplices. This means that the image of* ∂_1 *is the singleton set* $\{0\}$*.*

The zeroth homology class of K is $H_0(K) = ker\ (\partial_0)/Im\ (\partial_1) = C_0/\{0\}$*.*

The dimension of H_0 *is dim(ker* (∂_0)*) - dim(Im* (∂_1)*). This gives us* $4 - 0 = 4$*. So, the dimension of* $H_0(K)$ *is 4.*

Example 3.4.2. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_0, v_1\}, \{v_2, v_3\}.$$

The zero dimensional simplices in K include six elements, $\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}$ *and* $\{v_5\}$*. So, the vector space* $C_0 =< v_0, v_1, v_2, v_3, v_4, v_5 >$*. The kernel of* $\partial_0 = C_0$*.*

The one dimensional simplices in K include two elements, $\{v_0, v_1\}$ *and* $\{v_2, v_3\}$*. The vector space* $C_1 =< v_0v_1, v_2v_3 >$*. The images of these 1-dimensional simplices under* ∂_1 *are* $v_1 - v_0$ *and* $v_3 - v_2$ *respectively. So, the image of the* ∂_1 *is the vector space* $< v_1 - v_0, v_3 - v_2 >$*.*

The zeroth homology class of K is $H_0(K) = ker\ (\partial_0)/Im\ (\partial_1) = C_0/ < v_1 - v_0, v_3 - v_2 >$*.*

The dimension of H_0 *is dim(ker* (∂_0)*) - dim(Im* (∂_1)*). This gives us* $6 - 2 = 4$*. So, the dimension of* $H_0(K)$ *is 4.*

The first homology class of K is $H_1(K) = ker\ (\partial_1)/Im\ (\partial_2)$*. Here, both the ker* (∂_1) *and the Im* (∂_2) *are* $\{0\}$*. So,* $H_1 = \{0\}/\{0\}$*.*

The dimension of the first homology class of K. The dimension of H_1 *is* $0 - 0 = 0$*. So, the dimension of* $H_1(K)$ *is 0.*

Example 3.4.3. *Consider the simplicial complex*

$$K = \{\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\}, \{v_7\}, \{v_8\}, \{v_1, v_2\}, \{v_2, v_3\},$$

$$\{v_1, v_3\}, \{v_4, v_5\}, \{v_5, v_6\}, \{v_4, v_6\}, \{v_7, v_8\}, \{v_1, v_2, v_3\}\}.$$

The zero dimensional simplices in K include nine elements, $\{v_0\}, \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\}, \{v_7\}, \{v_8\}$. *So, the vector space* $C_0 =< v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8 >$. *The kernel of* $\partial_0 = C_0$.

The one dimensional simplices in K include seven elements, $\{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}, \{v_4, v_5\}, \{v_5, v_6\}, \{v_4, v_6\}, \{v_7, v_8\}$. *The vector space* $C_1 =< v_1v_2, v_2v_3, v_1v_3, v_4v_5, v_5v_6, v_4v_6, v_7v_8 >$.

The images of the basis elements of C_1 *is* $v_2 - v_1, v_3 - v_2, v_3 - v_1, v_5 - v_4, v_6 - v_5, v_6 - v_4, v_8 - v_7$. *The set Im* (∂_1) *is* $< v_2 - v_1, v_3 - v_2, v_5 - v_4, v_6 - v_5, v_8 - v_7 >$. *The natural question here is why the elements* $v_3 - v_1$ *and* $v_6 - v_4$ *are not included in the spanning set of Im* (∂_1). *This is because,* $v_3 - v_1$ *is the linear combination of* $v_2 - v_1$ *and* $v_3 - v_2$. *Similarly,* $v_6 - v_4$ *is a linear combination of* $v_5 - v_4$ *and* $v_6 - v_5$.

The zeroth homology class of K is $H_0 = ker\ (\partial_0)/Im\ (\partial_1) = C_0/ < v_2 - v_1, v_3 - v_2, v_5 - v_4, v_6 - v_5, v_8 - v_7 >$.

The dimension of H_0 *is* $\dim(ker\ (\partial_0)) - \dim(Im\ (\partial_1))$. *This gives us* $9 - 5 = 4$. *So, the zeroth homology of K is 4.*

Next, we need to find the ker (∂_1). *The ker* $(\partial_1) =< v_2v_3 - v_1v_3 + v_1v_2, v_5v_6 - v_4v_6, v_4v_5 >$. *The Im* (∂_2) *is the set* $< v_1v_2v_3 >$.

The first homology class of K is $H_1 = ker\ (\partial_1)/Im\ (\partial_2) =< v_2v_3 - v_1v_3 + v_1v_2, v_5v_6 - v_4v_6 + v_4v_5 > / < v_1v_2v_3 >$.

The dimension of H_1 *is* $\dim(ker\ (\partial_1)) - \dim(Im\ (\partial_2))$. *This gives us* $2 - 1 = 1$. *So, the first homology of K is 1.*

Homology classes tell us the number of holes of different dimensions. The zeroth homology of all the above examples is 4. All the examples have 4 components to them. The zeroth homology class tells us the number of connected components.

Through homology, we can find the number of holes in a geometric structure, thereby providing a necessary condition for two simplicial complexes to be termed equivalent. While this does not guarantee the equivalence of two structures, it is a very useful tool to differentiate the two structures based on the difference in their homology class. For example, a disk is definitely different from a doughnut since the former has no holes and the latter has one dimensional hole.

Chapter 4

Persistance Homology

4.1 Data as a Simplicial Complex

The idea of studying Topology over the course of this project is to be able to analyse data by studying the shape of the data. To be able to look at data geometrically, we create a simplicial complex with the different data points as the vertices of the simplicial complex. To convert the data into the simplicial complex, we define filtrations.

Definition 18. *Consider a simplicial complex K. A filtration of K (of length n) is a nested sequence of sub-complexes of the form:*

$$F_1K \subset F_2K \subset \cdots \subset F_{n-1}K \subset F_nK$$

where $F_{i-1}K \neq F_iK$ for all i.

Example 4.1.1. *Consider the simplicial complex*

$$K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_2, v_4\}, \{v_1, v_4\}, \{v_1, v_3\}, \{v_1, v_4\}\}.$$

The following nested sequence of sub-complexes forms a filtration of K of length 4.

$F_1K = \{\{v_1\}, \{v_2\}\}$

$F_2K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\}$

$F_3K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_2, v_4\}, \{v_1, v_4\}\}$

$F_4K = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_1, v_2\}, \{v_2, v_4\}, \{v_1, v_4\}, \{v_1, v_3\}, \{v_1, v_4\}\}$

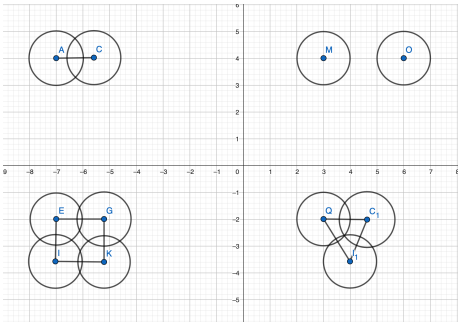
This is a filtration of K that starts with 2 vertices and has $F_1K \subset F_2K \subset F_3K \subset F_4K = K$.

The Cech Complex and the Vietoris Rips Complex are two such filtrations that are popularly used to create a simplicial complex from a given set of points. For both these filtrations, we need to fix a parameter R.

4.1.1 Cech Complex

The Cech complex is an abstract simplicial complex that we construct from a point cloud in the following way. Let us denote the Cech Complex as $\mathcal{C}_R$.

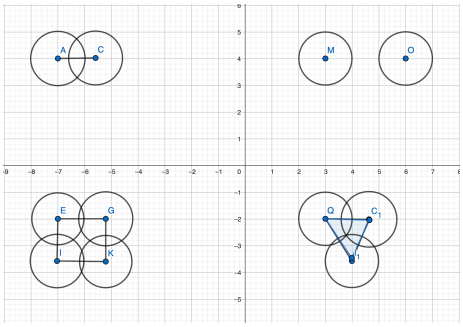
Let us consider a data set X that contains points in $\mathbb{R}^k$. We fix a parameter R. We draw a ball of radius R around each of these data points. Upon drawing these balls, if two balls intersect each other, we draw an edge between the two vertices that form the centre of the two respective balls. If the intersection of three balls is non-empty, then we draw a shaded triangle with the three data points as vertices of the triangle. Generally, if the intersection of k balls is non-empty, then a $k-1$ simplex is formed. The highest dimension simplex that can be drawn from n data points is n.



4.1.2 Vietoris-Rips Complex

The Vietoris-Rips complex is an abstract simplicial complex that we construct from a point cloud in the following way. Let us denote the Vietoris-Rips Complex as $\mathcal{V}_R$. Let us consider a data set X that contains points in $\mathbb{R}^k$. We fix a parameter R. We draw balls of radius R

around each of the data points. Upon drawing these balls, if two balls intersect each other, we draw an edge between the two vertices that form the centre of the two respective balls. If three balls mutually interact, a filled triangle is drawn with the three centres being the three vertices. Generally, if k balls mutually intersect, then a $k-1$ simplex is formed.



If we look at the two images, we will see that the triangle QC_1V_1 is not shaded in the Cech complex while the same triangle is shaded in the Vietoris-Rips complex.

4.2 Persistence homology

The Cech and Vietoris-Rips complexes are constructed by considering a parameter R. For which R should we be studying these properties of the complex? We do not study the complex for a fixed radius R. We take the properties of the simplicial complexes that are formed for a range of R. We vary the value of R while constructing the complexes and thereby get different Cech and Vietoris-Rips complexes. As we vary R, the homology of these complexes will also change. For each value of k , we record the both and death of the generators of H_k as we vary R. As the generators completely describe H_k, this is good enough.

The $\dim(H_0)$ will give us the number of connected components, the $\dim(H_1)$ will tell us whether there is a one-dimensional loop or hole that is not filled, and so on.

The complexes constructed for each value of R gives us some information about the geometry of the complex. Properties that persist for a long period of time, that is, over a large range of R, are called features of data. If the properties do not persist, we ignore them as noise.

We record the value of R where a hole is born and dies. We capture the birth and death of

data using a persistence diagram.

We have written a code to compute the homology of a Vietoris-Rips complex. The data points are of two kinds, one that is a set of random points that form a square and another that is a set of random points that form a square with a circular cutout in the center.

The following block of code finds the data points with random points in a square and its persistence diagram.

```
no_cut <- function(n)
{
  i=0
  x <- vector(mode='list', length=n)
  y <- vector(mode='list', length=n)
  while(i <= n)
  {
    a <- runif(1, -3, 3)
    b <- runif(1, -3, 3)
    x[i] <- a
    y[i] <- b
    i=i+1
  }
  print(x)
  c <- append(x,y)
  p2 <- matrix(unlist(c), ncol = 2,nrow=n)
  plot(x, y)
  return(p2)
}
nocut_1 = no_cut(500)
Diag <- ripsDiag(X=nocut_1, maxdimension = 1, maxscale = 5, printProgress = FALSE)
plot(Diag[["diagram"]])
```

The following block of code finds the data points with random points in a square with a circle cutout in the center and its persistence diagram.

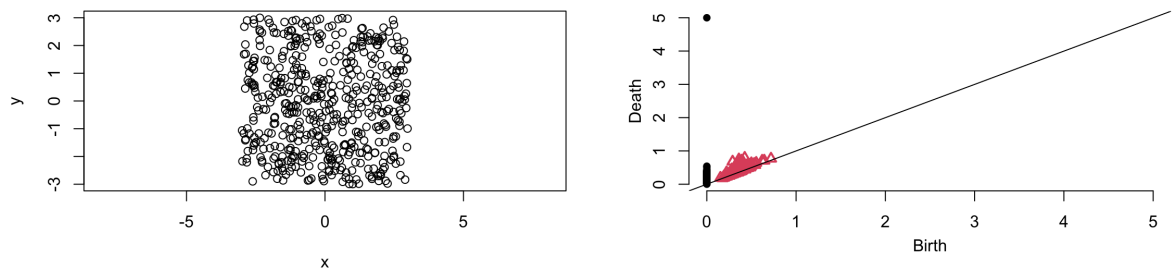


Figure 4.1: A data set consisting of random points in a square and the persistence diagram of this data set.

```
cutout <- function(n)
{
  i=0
  x <- vector(mode='list', length=n)
  y <- vector(mode='list', length=n)
  while(i <= n)
  {
    a <- runif(1, -3, 3)
    b <- runif(1, -3, 3)
    if(a**2 + b**2 > 1)
    {
      x[i] <- a
      y[i] <- b
      i=i+1
    }
  }
  print(x)
  c <- append(x,y)
  p2 <- matrix(unlist(c), ncol = 2,nrow=n)
  plot(x, y)
  return(p2) }
```

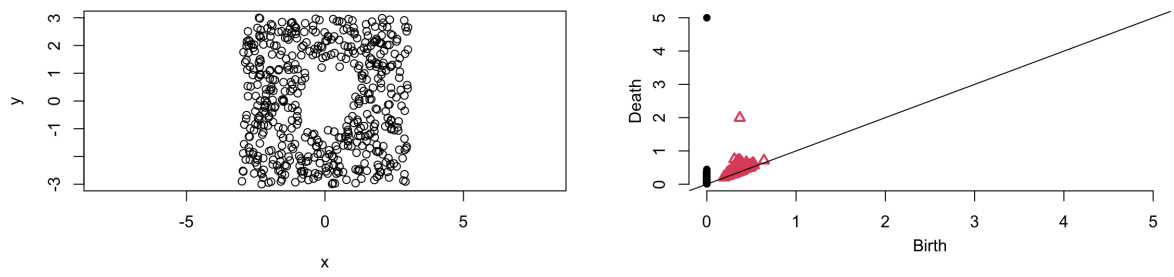


Figure 4.2: A data set consisting of random points in a square with a circular cutout in the center and the persistence diagram of this data set.

```
cutout_1 <- cutout(500)
Diag <- ripsDiag(X=cutout_1, maxdimension = 1, maxscale = 5, printProgress = FALSE)
plot(Diag[["diagram"]])
```

We use these persistence diagrams to distinguish between data. We say that two data sets are similar when they have very similar persistence diagrams. These persistence diagrams can be thought of as a proxy for the actual data sets. We gauge the similarity of the persistence diagrams by computing the Wasserstein's distance between the two diagrams.

4.3 Wasserstein's distance

The Wasserstein's distance is a measure used to understand the similarity between two persistence diagrams.

Definition 19. *The Wasserstein's distance between two persistence diagrams X and Y is*

$$W_p(X,Y) = [\inf_{\eta} \sum_{x \in X} ||x - \eta(x)||_{\infty}^{p}]^{1/p}$$

where η is all possible bijections between X and Y.

This distance will tell us how similar two point clouds are and thereby, tell us how similar two data sets are.

In the next chapter, we will be looking at an application where this distance has been used to classify Karate moves.

Chapter 5

Classification of Karate moves using Topological Data Analysis

After studying the concepts of topology required to analyse data sets using the method of Topological Data Analysis or TDA, we started to look through potential applications in the fields of engineering, bio-medicine and chemistry, to name a few. The field of computer vision had a lot of promising research that was very interesting. Computer vision is a field of artificial intelligence that studies and classifies digital images and videos. Most of the research in this field involves using machine learning to perform the classification. But to study the data through machine learning, we would need a very large set of data. Often, the data sets are small and therefore, machine learning algorithms are not the ideal way to classify such data sets. Topological Data Analysis is a novel way to analyse data by using techniques from topology. Researchers have been studying computer vision through TDA. So, we decided to explore the same.

The data sets were from varying situations like home safety footage, bio-medical imaging, neuro-imaging and so on. We studied the use of Topological Data Analysis in human motion recognition. The paper that we followed was "The application of topological data analysis to human motion recognition" by Marcin Żelawski and Tomasz Hachaj Żelawski and Hachaj 2021. The authors are Polish researchers and the paper was published in the year 2021.

Through this chapter, I aim to first provide a brief explanation of Principal Component Analysis (PCA) and a summary of the paper. Further, I will provide a run-through of the algorithm used by the authors by elucidating the algorithm by including compatible sections of code. The code has been provided by the authors along with the article.

5.1 Principal Component Analysis

Principal Component Analysis is a machine learning method that is used to reduce the dimension of large data sets. The paper, "The application of topological data analysis to human motion recognition," claims that TDA outperforms PCA-based feature recognition. So, we first understand PCA.

If we have only one variable, we can plot the data on a number line. If we have two variables, we can plot the data on a plane. If we have three variables, we would have another axis and plot the data on a three-dimensional space. Once we reach four variables, we can no longer plot them on a scatter plot. Through PCA, we take higher-dimensional data and plot it on a lower-dimensional plane.

Let us assume that we have a scatter plot of three dimensional data. We first calculate the average value for each of the three variables. We plot this point and call this the centre of the data. We shift the data points such that the centre coincides with the origin but the position of the data points respective to each other remains the same.

Next, we want to find the best fit line for this data set. How do we know which is the best-fit line? We draw a line through the origin and project the data points onto this line. We then rotate this line until the sum of the squared distances of the projected points to the origin is maximised. We square the distances so that the negative values do not cancel out the positive values. This line is the best-fit line and is called the Principal Component 1 or PC1. This line is a combination of the three variables.

We then calculate the unit vector along the PC1 and call this the Singular Vector. This gives us the proportion with which the variables are combined. These proportions with which the three variables make up the PC1 are called the loading scores.

While computing the best fit line, we maximised the sum of the squared distances of the projected points to the origin. Now, if we take the square root of the sum of the squared distances of the projected points to the origin, we call this the Singular Value.

Now that we have covered Principal Component 1, let us move onto Principal Component 2. The line through the origin that is perpendicular to PC1 and is the best-fit line is called the Principal Component 2 or PC2.

Next, we take PC1 to be the x-axis and PC2 to be the y-axis. We project the points onto PC2 as well. Finally, we project the points back onto the plane by taking the value on PC1 as the x-coordinate and the value on PC2 as the y-coordinate. The sum of the squared distances of the projected points on PC1 to the origin divided by $n - 1$, where n is the number of data points, is called the variation of PC1. The same calculation is done for PC2.

We can also find a PC3, the line that is perpendicular to both PC1 and PC2 and passes through the origin. But, we want to achieve dimension reduction.

We get as many Principal Components as there are variables in the data set. But the first two PCs carry a large part of the information regarding the variation in the data points. So, plotting just the PC1 and PC2 would be enough for recognising clusters in the data. Even if the first two PCs do not carry most of the variation, we will still be able to identify clusters and therefore, get information from just the first two PCs.

So, even though we can get as many PCs as there are variables, we only consider the first two PCs for plotting the data. This way we convert a higher dimensional data set into a 2D plot with PC1 as the x-axis and PC2 as the y-axis.

5.2 Summary of "The application of topological data analysis to human motion recognition"

One of the areas where motions capture data is used extensively is in the field of sports. The paper "The application of topological data analysis to human motion recognition" uses Topological Data Analysis for motion capture processing and human action recognition in Karate. Human capture recognition is called MoCap (motion capture).

The authors use an open-source data set of three Shorin-Ryu karate players. They have been recorded while performing twelve types of Karate techniques, each repeated ten times. So, the data set contains 360 motion capture data points. The data was captured using systems involving seventeen Internal Measurement Unit (IMU) sensors. The motion recordings contain twenty body joints.

The twelve moves have been performed with high speed and precision. Due to the complexity of the moves, there are limited data sets available for analysis. This makes machine learning hard on these data sets.

First, a broad analysis was done on the data before TDA. The data were classified broadly into bounding boxes. If the bounding boxes of the Karate moves matched, a further TDA was performed to precisely classify the moves. This was to increase efficiency. Not every move had to be verified with every other move in the TDA stage if the bounding boxes did not match. The full algorithm will be discussed in the next section.

The algorithm was implemented using R. The data sets were split such that the data from two players were used as training data sets and the data from the third player was used as the validation data set. A three-way cross-verification was performed where all possible combinations were taken. The results were averaged. The topological method gave a recognition rate of 0.975.

There were instances where the left and right versions of the same move were misclassified. This is because Topology does not efficiently differentiate such a difference as it is stable to rotations, translations and scaling.

The PCA-based method gave a recognition rate of 0.647 with a similar three-fold cross-verification. This proves that TDA-based analysis outperforms PCA-based analysis and is therefore an efficient method for human action recognition.

5.3 Algorithm used in "The application of topological data analysis to human motion recognition"

The paper "The application of topological data analysis to human motion recognition" has effectively analysed and evaluated the novel method of Topological Data Analysis to classify

Karate moves. They have established that it is a better classifier than PCA for human motion recognition.

In this section, we explore the algorithm that the authors have devised to perform the aforementioned TDA classification. The algorithms is two-fold. In the first step, the authors are performing a crude classification of the different moves. This is called the bounding box stage. The next step uses TDA. Those point clouds that clear the bounding box stage are analysed using 0th-dimensional persistence homology. The algorithm was implemented using R. Let us first define the terminology used in the paper,

- M represents the number of sensors.
- i represents the co-ordinates and can take the value of 0, 1 or 2.
- s_{ij} is the position of the i-th co-ordinate of the j-th sensor.
- K represents the number of measurements.

Now, let us consider the set $v_k = \{t_k, s_{01k}, s_{11k} \dots s_{2Mk}\}$. The point cloud will be the set $\{v_k \in \mathbb{R}^{3M+1} | k = 1, \dots K\}$. This point cloud is transformed into a Vietoris-Rips complex and the persistence homology is visualised using persistence diagrams.

5.3.1 Bounding boxes

The bounding boxes are constructed around the data points as a first level of filtering. Each Karate athlete had sensors on twenty body joints. For this stage of analysis, the hip sensor is taken as the origin.

We take two Karate athletes as the training data set and the third Karate athlete as the validation data set. For the point clouds representing the different moves, the minimum and maximum co-ordinates reached by the j-th sensors or the maximal and minimal co-ordinates of the bounding boxes of the j-th sensors were calculated with respect to the hip sensor. It is computed as follows:

$$x_j^{\min} = (min_k\ s_{0jk}, min_k\ s_{1jk}, min_k\ s_{2jk}) - (s_{0h0}, s_{1h0}, s_{2h0}).$$

$$x_j^{\max} = (max_k\ s_{0jk}, max_k\ s_{1jk}, max_k\ s_{2jk}) - (s_{0h0}, s_{1h0}, s_{2h0}).$$

```
pc <- PointCloud
    BoxList <- list()

      # LeftHand
      min <- c(min(pc[,"lhand_x"]), min(pc[,"lhand_y"]), min(pc[,"lhand_z"]))
      max <- c(max(pc[,"lhand_x"]), max(pc[,"lhand_y"]), max(pc[,"lhand_z"]))
      Box <- rbind(min, max)
      BoxList <- append(BoxList, list(Box))
      # RightHand
      min <- c(min(pc[,"rhand_x"]), min(pc[,"rhand_y"]), min(pc[,"rhand_z"]))
      max <- c(max(pc[,"rhand_x"]), max(pc[,"rhand_y"]), max(pc[,"rhand_z"]))
      Box <- rbind(min, max)
      BoxList <- append(BoxList, list(Box))
      # LeftFoot
      min <- c(min(pc[,"lfoot_x"]), min(pc[,"lfoot_y"]), min(pc[,"lfoot_z"]))
      max <- c(max(pc[,"lfoot_x"]), max(pc[,"lfoot_y"]), max(pc[,"lfoot_z"]))
      Box <- rbind(min, max)
      BoxList <- append(BoxList, list(Box))
      # RightFoot
      min <- c(min(pc[,"rfoot_x"]), min(pc[,"rfoot_y"]), min(pc[,"rfoot_z"]))
      max <- c(max(pc[,"rfoot_x"]), max(pc[,"rfoot_y"]), max(pc[,"rfoot_z"]))
      Box <- rbind(min, max)
      BoxList <- append(BoxList, list(Box))
```

The above code performs this action three times, once for each point cloud.

The next step is to compute the distance between the bounding boxes of the training point cloud C_1, and the validation point cloud C_2, for the j-th sensor. We calculate this distance by summing the distance between the minimal co-ordinates of the j-th sensor in the two point clouds and the distance between the maximal co-ordinates of the j-th sensor in the two point clouds. This is computed as follows:

$$d_j^{box}(C_1, C_2) = d(x_j^{\min}(C_1), x_j^{\min}(C_2)) + d(x_j^{\max}(C_1), x_j^{\max}(C_2))$$

where d refers to the Euclidean distance between the two points.

We now define the hand foot distance for each of the athletes respectively. We calculate this distance to be the sum of the distances between the bounding boxes of the left hand sensor, right hand sensor, left foot sensor and right foot sensor, of the training and the validation point clouds. This is computed as follows:

$$d^{hf}(C_1, C_2) = d_{hl}^{box}(C_1, C_2) + d_{hr}^{box}(C_1, C_2) + d_{fl}^{box}(C_1, C_2) + d_{fr}^{box}(C_1, C_2).$$

The value of d^{hf} measures the similarity between the bounding boxes of the two point clouds. This value detects the differences between the moves very quickly. So, only those point clouds that are not rejected by the bounding box algorithm are analysed using TDA in the next stage.

The box distance function has been implemented as follows.

```
box_distance <- function(bl_1, bl_2)
{
  box_dist <- 0

  l1 <- length(bl_1)
  l2 <- length(bl_2)

  if( l1 == l2 & l1 > 0)
  {
    for(i in 1:l1)
    {
      n1 <- length(bl_1[[i]][,1])
      n2 <- length(bl_2[[i]][,1])

      if( n1 == n2 & n1 > 0)
      {
```

```
        for(j in 1:n1)
        {
          box_dist <- box_dist + euc_dist(bl_1[[i]][j,], bl_2[[i]][j,])
        }
      }
    }
  }
  return( box_dist )
}
```

Once the box distance has been computed, they create an array with the box distance. This is done as follows.

```
Movement <- list(FileName = FileName, MovName = MovName, File = File,
MovName_1 = MovName_1, MovName_2 = MovName_2,
PointCloud = PointCloud, Diag.info = Diag.info, BoxList = BoxList)
MovementList <- append( MovementList, list(Movement) )
```

5.3.2 Topological Data Analysis

Once the bounding box stage is complete, those point clouds that were rejected during the bounding box stage are removed from the training data set. We now have a smaller list of point clouds that we need to compare the point clouds in the validation data set with.

To perform Topological Data Analysis, we first convert the point clouds into Vietoris-Rips complexes. The authors only calculate the 0 dimensional persistence homology. By varying R, we get the generators of H_0 at different values of R. The values at which the generator or 0 dimensional holes are born and die are recorded. This is converted into a persistence diagram. The Wasserstein's distance between the persistence diagrams of the training and validation data sets is calculated.

This process is done by taking all possible combinations of two Karate athletes as the training data set and the third Karate athlete as the validation data set, thereby, performing the three-way cross verification process.

The parameter ϵ is set empirically. The algorithm returns a unique match in the training data set for the point cloud in the validation data set.

The following block of code finds the minimum distance between the persistence diagrams between the training and the validation data set.

```
if( TDA_ON > 0 )
  {
    rec_dist_sort <- rec_dist[order(sapply(rec_dist, '[[', "Dist"))]
    if( rec_dist_sort[[2]]$Dist - rec_dist_sort[[1]]$Dist <= TDA_VERIF_MARGIN )
    {
      min_dist_final <- Inf
      min_idx_final <- -1
      for (i in 1:min(TDA_VERIF_SIZE, length(rec_dist_sort)))
      {
        PatIndx <- rec_dist_sort[[i]]$Indx
        PatMov <- MovementListPat[[ PatIndx ]]
        Diag2 <- PatMov$Diag.info$diagram
        Mov <- MovementList[[m1]]
        Diag.info <- ripsDiag(X=Mov$PointCloud, maxdimension=0,
        maxscale=TDA_MAXSCALE,
        dist="euclidean", TDA_library, printProgress=FALSE)
        Diag1 <- Diag.info$diagram

        distance_TDA <- wasserstein(Diag1 = Diag1, Diag2 = Diag2,
        dimension = d)

        # find the move with minimum wasserstein distance from m1
        if( distance_TDA < min_dist_final )
        {
          min_dist_final <- distance_TDA
          min_idx_final <- PatIndx
        }
```

```
        }
        #we check if for the training data set, the movement names
        #at the the two indices match or not if they do not match for
        #the training data set, probably TDA is not helping us, thus we go
        #back to default classification using box distance.

        if( min_idx_final > -1 & min_idx_xyz > -1 )
        {
          name_xyz <- MovementListPat[[min_idx_xyz]]$MovName_1
          name_xyz_s <- MovementListPat[[min_idx_xyz]]$MovName_2
          name_TDA <- MovementListPat[[min_idx_final]]$MovName_1
          name_TDA_s <- MovementListPat[[min_idx_final]]$MovName_2
          if( name_xyz != name_TDA & name_xyz_s == name_TDA_s )
          {
            min_dist_final <- distance_xyz
            min_idx_final <- min_idx_xyz
          }
        }
      }
    }
```

Once the index with the minimum Wasserstein distance between the persistence diagrams of the training and validation set is retrieved, we check if the name of the move at that index matches the name of the move on the validation data set. If it does, we increase the counter by one.

```
    if( min_idx_final > -1 )
      {
        dist_final[d+1, m1, 2] <- MovementListPat[[min_idx_final]]$MovName

        mov_name_1 <- MovementList[[m1]]$MovName_1
        mov_name_2 <- MovementListPat[[min_idx_final]]$MovName_1
```

```
if( mov_name_1 == mov_name_2 )
  good_rec_final <- good_rec_final + 1
  dist_final[d+1, m1, 1] <- "OK"
else
{
   dist_final[d+1, m1, 1] <- "ERR"
}
```

The classification rate for the algorithm is 0.975.

Chapter 6

Conclusion

The aim of the project was to study and analyse the shape of data and explore an application of Topological Data Analysis. During the first half of the project, we studied topology to be able to study the shape of data. Once the theoretical study was done, we explored the different fields where TDA is applied. Some of these fields included engineering, bio-medicine and chemistry. One of the most interesting areas we discovered was computer vision. So, we read and analysed the paper, "The application of topological data analysis to human motion recognition" written by Marcin Żelawski and Tomasz Hachaj.

The paper provided a topological way to classify Karate moves. We also explored and understood the R code that the authors had supplemented the paper with. The code implemented the algorithm developed in the paper.

Upon understanding the code, we wanted to try and analyse a different data set using the method explicated in the paper by Żelawski and Hachaj. We came across the paper, "Fine Classification of Complex Motion Pattern in Fencing." Mantovani et al. 2010 The aim of the paper was to classify the movements of fencers from motion capture data using PCA, wavelet-based analysis and a feature extraction method.

Since the paper uses PCA as a first step, we wanted to understand if TDA provided a better classification. At first glance, the data sets looked very similar, thereby piquing our interest. However, the data sets provided by the authors were single dimensional, that is, PCA had

already been performed on the data set. We were unable to procure the raw data sets and where therefore, unable to perform our own analysis.

There are many fields where TDA is now being applied to. It is an interesting way to look at data and analyse it thereby, increasing its popularity amongst all researchers.

Bibliography

Nanda, Vidit (n.d.). *Computational Algebraic Topology: Lecture notes*. https://people.maths.ox.ac.uk/nanda/cat/TDANotes.pdf (cit. on p. 3).

Żelawski, Marcin and Tomasz Hachaj (2021). "The application of topological data analysis to human motion recognition". In: *Technical Transactions* 118.1 (cit. on p. 36).

Mantovani, G et al. (2010). "Fine classification of complex motion pattern in fencing". In: *Procedia Engineering* 2.2, pp. 3423–3428 (cit. on p. 47).