DEGREE PROJECT IN APPLIED AND COMPUTATIONAL MATHEMATICS
120 CREDITS, SECOND CYCLE
STOCKHOLM, SWEDEN 2016

Illustrations of Data Analysis Using the Mapper Algorithm and Persistent Homology

RAMI KRAFT

KTH ROYAL INSTITUTE OF TECHNOLOGY

SCHOOL OF ENGINEERING SCIENCES

Illustrations of Data Analysis Using the Mapper Algorithm and Persistent Homology

RAMI KRAFT

Master's Thesis in Mathematics (30 ECTS credits)
Master Programme in Applied and Computational Mathematics (120 credits)
Royal Institute of Technology year 2016
Supervisor at KTH: Wojciech Chacholski
Co-supervisor: Ryan Ramanujam (KI)
Examiner: Wojciech Chacholski

TRITA-MAT-E 2016:03
ISRN-KTH/MAT/E--16/03--SE

Royal Institute of Technology
SCI School of Engineering Sciences

KTH SCI
SE-100 44 Stockholm, Sweden

URL: www.kth.se/sci

"Algebra is generous; she often gives more than is asked for. "

Jean d'Alembert (1717-1783)

TH ROYAL INSTITUTE OF TECHNOLOGY

Abstract

Illustrations of Data Analysis Using the Mapper Algorithm and Persistent Homology

by Rami KRAFT

The Mapper algorithm and persistent homology are topological data analysis tools used for analyzing point cloud data. In addition a classification method is used as a part of the data analysis toolchain adopted in this thesis in order to serve as a distinguishing technique for two class labels.

This thesis has two major goals; the first goal is to present persistent homology and the Mapper algorithm as two techniques by which shapes, mostly point clouds sampled from shapes of known topology can be identified and visualized even though in some cases noise is being there. We then provide some illustrative examples in the form of barcodes, persistence diagrams and topological network models for several point cloud data.

The second goal is to propose an approach for extracting useful insights from point cloud data based on the use of Mapper and a classification technique known as the penalized logistic regression. We then provide two real-world datasets for which both continuous and categorical responses are considered. We show that it is very advantageous to apply a topological mapping tool such as the Mapper algorithm on a dataset as a pre-processing organizing step before using a classification technique.

We finally show that the Mapper algorithm not only allows for visualizing point cloud data but also allows for detecting possible flare-like shapes that are present in the shape of the data. Those detected flares are given class labels and the classification task at that point is to distinguish one from the other in order to discover relationships between variables in such a way that allows for generalizing those relationships to hold on previously unseen data.

Sammanfattning

Mapper-algoritmen och persistent homologi tillämpas som verktyg inom topologisk dataanalys. Dessutom används en klassiceringsmetod för att kunna skilja mellan två klasser som definieras i en topologisk nätverksmodell.

Syftet är framförallt att uppnå två huvudmål; det första att presentera persistent homologi och Mapper-algoritmen som verktyg för att identifiera och visualisera punktmoln samplade från objekt av känd topologi även i sådana fall där slumpmässiga fel förekommer. Detta illustreras med flera exempel av punktmoln och motsvarande persistenta diagram samt topologiska nätverksmodeller.

Det andra målet är att föreslå en metod baserad på användning av Mapper-algoritmen och en klassificeringsteknik för att kunna extrahera kunskap ur två datamängder, varav den ena har en kategorisk responsvariabel och den andra en kontinuerlig.

Slutligen visas att Mapper-algoritmen gör det möjligt att omvandla högdimensionell data till en tvådimensionell graf som är lätt att visualisera och som sedan kan analyseras genom statistiska inlärningsmetoder.

Acknowledgements

I would like to thank Professor Wojciech Chachólski and Postdoc Ryan Ramanujam for providing the support for this master thesis; interesting discussions and beneficial comments are acknowledged and very much appreciated…

Contents

1. Scope and background

Over the last decade the amount of collected data has increased intensely. Many datasets are typically large, noisy or complex; for that reason extracting useful insights can be a real challenge.

Persistent homology and the Mapper algorithm are tools used in topological data analysis to study and better understand point cloud data. The basic motivation behind topological data analysis is that data has shape (Carlsson, 2009).

1.1 Types of data

There are different types of data; mostly we deal with nominal categorical data that is data containing unordered qualitative information, for example M=male or F=female. Ordinal categorical data are different in the sense that they are ordered, for example 1=cold, 2=medium and 3=hot.

There is also another type of data known as count data; those are quantitative and discrete, for example the number of children a couple has. Ratio scale data are quantitative but continuous and there is an absolute zero, for example weight or temperature in Kelvin.

1.2 Classification and clustering

In general classification systems are either supervised or unsupervised. A supervised classification algorithm learns from a training data sample that has been manually classified into a finite number of categories and the goal is to define rules that can classify new incoming data points. There are several approaches that can be used for supervised classification, for example linear or quadratic discriminant analysis as well as the penalized logistic regression which we are going to be focusing on.

In unsupervised classification or clustering no labelled data are available; instead the goal is to organize observations into clusters based on a notion of similarity. There are many hierarchical clustering algorithms including single linkage, complete linkage, average linkage or centroid linkage. As we shall see later, clustering is an important component of the Mapper methodology.

1.3 Topological data analysis

Topological data analysis (TDA) is a growing branch of applied mathematics that among others uses ideas from algebraic topology and the goal is to be able to analyze complex high dimensional data.

Since the basic motivation behind TDA is that data has shape, we therefore study the shape of data and try to extract invariant topological features that might help us discover relationships and patterns in data.

There are three key ideas from topology that makes extracting knowledge from data through shape possible and in some sense better than standard data analysis, see Figure 1.1.

Those key ideas are:

1) *Coordinate invariance:* No dependence on a coordinate system. Suppose we are interested in identifying the loop shown in Figure 1.1 (left). If the coordinates are for example stretched out, still there will be no effect on our ability to identify the loop. One benefit is that we are able to analyze data collected from different platforms that may use different technologies.

2) *Deformation invariance:* That is to say a shape such as the letter A has one loop and this property of the letter A does not change despite stretching or squashing the letter as long as we are not going to tear it apart, see Figure 1.1 (middle). This implies less sensitivity to noise.

3) *Compressed representation:* We can approximate, for example, an object made of thousands of points or even infinitely many points like a complete circle using only some vertices and edges, see Figure 1.1 (right).

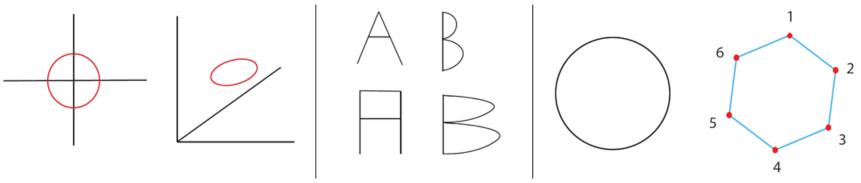

FIGURE 1.1: Three key ideas from topology: Coordinate freeness (left), invariance under deformation (middle) and compressed representations (right) (Carlsson, 2009).

2. Persistent homology and Mapper

Topology is the branch of mathematics concerned with the study of shapes up to different notions of equivalence. Consider for example two objects made of elastic rubber; one can make them equivalent if one of the two objects can be deformed into the other by stretching or squashing as long as we are not going to tear the object apart. A famous example is deforming a donut into a coffee cup, see Figure 2.1. Nevertheless the idea of topological equivalence is not restricted to physical spaces. Algebraic topology, however, is concerned with solving topological problems using algebraic methods. A topological space can be triangulated by means of simplicial complexes which are built up from points, line segments, faces and higher dimensional analogues.

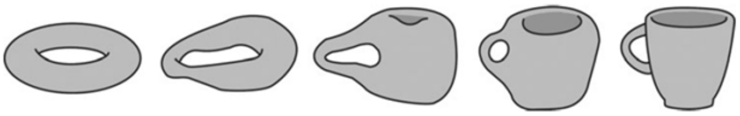

FIGURE 2.1: A transformation from donut to coffee cup.

2.1 Some basic definitions from topology

One of the most important problems in topology is to classify topological spaces up to topological or homotopical equivalence. In order to show that two topological spaces are not equivalent, one needs to find topological invariants that distinguish one from the other since if the two topological spaces have different invariants then they cannot be homeomorphic or homotopic.

2.1.1. Recall that a topological space is a pair of (X, τ), where τ is the topology of (X, τ) consisting of a family of subsets of X i.e. $\tau \subseteq 2^X$ such that:

1. X and $\emptyset$ are elements of τ i.e. X,$\emptyset \in \tau$.

2. Union of member of τ are in τ i.e. if $A_1, \dots, A_n$ are elements of τ, then their unions $A_1 \cup A_2 \cup \dots \cup A_n \in \tau$.

3. Finite intersections of members of τ are in τ i.e. if $A, B \in \tau$ then $A \cap B \in \tau$.

A topological space X is said to be a Hausdorff space if given any pair of distinct points $p_1, p_2 \in X$, there exist neighborhoods U_1 of p_1 and U_2 of p_2 with $U_1 \cap U_2 = \emptyset$ i.e. points can be separated by open subsets.

2.1.2. Let d be a metric on a set X. Using this metric we define a topology on X by declaring a subset $U \subset X$ to be open if for any $x \in U$ there is a positive real number ϵ such that $\{y \in X \mid d(x, y) < \epsilon\}$ belongs to U. Such topological space is called a metric space and any metric space is Hausdorff.

2.1.3. Here we recall various ways of measuring distances on the set of n-tuples of real numbers:

- **L1.** $\|x, y\|_1 := \sum_{i=1}^{n} |x_i - y_i|$.
- **L2.** $\|x, y\|_2 := \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$.
- **Chebyshev.** $d(x, y) := \max\limits_{1 \le i \le n} |x_i - y_i|$.

All the above metrics induce the same topology on the set of n-tuples of real numbers. The obtained topological space we denote by $\mathbb{R}^n$ and call it the Euclidean space. We use the symbol $[0, 1]$ to denote the topological space given by the subset $\{x \in \mathbb{R} \mid 0 \le x \le 1\}$ of $\mathbb{R}$ with the topology induced by any of the above metric.

We finally mention that there are many other ways of measuring distances between n-tuples of real numbers. Some of them may lead to different topologies than the Euclidean one; for example the hamming distance. For more details about other metric spaces and different metrics on n-tuples of real numbers we refer the reader to (A. Babu, 2013).

2.1.4. Let X and Y be topological spaces. Their product is a topological space $X \times Y$ whose underlying set is the usual cartesian product with the topology defined as follows: a subset $U \subset X \times Y$ is open if for any $(x, y) \in U$, there are open subsets $x \in V \subset X$ and $y \in W \subset Y$ such that $V \times W \subset U$.

2.1.5. A function or a transformation from a set X to a set Y is denoted by $f : X \to Y$. A composite of functions $f : X \to Y$ and $g : Y \to Z$ is a function $g \circ f : X \to C$ defined by $g \circ f(x) = g(f(x))$. A function f is one-to-one if $f(x) = f(y)$ implies that $x = y$ while f is onto if for every $y \in Y$ there is $x \in X$ with $f(x) = y$. A function f is a bijection if it is both one-to-one and onto, in which case there is a function $f^{-1} : Y \to X$ such that $f \circ f^{-1}$ and $f^{-1} \circ f$ are the identity functions. The function f^{-1} is called the inverse of f.

Let X and Y be topological spaces. A function $f : X \to Y$ is called continuous if the inverse image of an arbitrary open set in Y is open in X. A continuous function is also called a map. A map $f : X \to Y$ is called a homeomorphism if it is a bijection and the inverse function $f^{-1} : Y \to X$ is continuous.

Two topological spaces X and Y are said to be isomorphic or homeomorphic if there is a a homeomorphism $f : X \to Y$ which is the most fundamental notion of topological equivalence, for example the sphere and the surface of tetrahedron are homeomorphic. Note also that $\mathbb{R}^2$ is homeomorphic to the product $\mathbb{R} \times \mathbb{R}$.

2.1.6. Another more flexible notion of equivalence is homotopy equivalence. Recall that two maps $f, g : X \to Y$ are called homotopic written as $f \simeq g$ if one can be continuously deformed into the other; explicitly if there is a map $h : X \times [0, 1] \to Y$ such that $h(x, 0) = f(x)$ and $h(x, 1) = g(x)$ for any x in X. A map $f : X \to Y$ is called a homotopy equivalence if there is a map $g : Y \to X$ such that $f \circ g$ and $g \circ f$ are homotopic to the identity maps. Finally two spaces X and Y are called homotopy equivalent if there is a homotopy equivalence $f : X \to Y$.

All spaces that are homeomorphic are also homotopy equivalent however the converse is not true. For example, the circle and the annulus are homotopy equivalent but are not homeomorphic since there is no continuous bijection between them.

2.2 Simplices

A k-dimensional simplex or k-simplex in $\mathbb{R}^n$ is defined in terms of $k + 1$ linearly independent points in $\mathbb{R}^n$. Let $v_0, \ldots, v_k$ be linearly independent points in $\mathbb{R}^n$. This by definition means that the vectors $v_0v_1, v_0v_2 \ldots, v_0v_k$ are linearly independent. The k-simplex denoted by $[v_0, \ldots, v_k]$ is the topological space given by the set

$$\left\{ \sum_{i=0}^{k} t_i v_i \,\middle|\, t_0 + t_1 + \cdots + t_k = 1, t_i \geq 0 \right\}$$

with the topology induced by the Euclidian metric. The numbers t_i are the coordinates of the point $x = \sum\limits_i t_i v_i \in [v_0, \ldots, v_k]$. Note that any two k-simplices are homeomorphic.

The 0-simplex consists of one point or a vertex. The 1-simplex can be written as $\{ t_0 v_0 + t_1 v_1 | \, t_0 + t_1 = 1, t_{0,1} \geq 0 \} = \{ t_0 v_0 + (1 - t_0) \, v_1 \}$ which is a segment or an edge with end points v_0 and v_1. Analogously the 2-simplex is a triangle with vertices v_0, v_1 and v_2 together with the three edges $v_0 + v_1$, $v_1 + v_2$ and $v_0 + v_2$, see Figure 2.2.

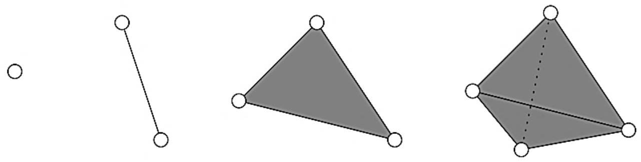

FIGURE 2.2: From left to right respectively are Simplices of dimension zero, one, two and three.

2.3 Simplicial complexes

Let X be a finite set. A simplicial complex K on X is a set of subsets of X such that:

- $\{x\} \in K$ for any x in X.
- if $\sigma \in K$ and $\tau \subset \sigma$, then $\tau \in K$.

An element $\sigma \in K$ is called a simplex of dimension $|\sigma| - 1$ where $|\sigma|$ denotes the number of elements in the set σ. We use the symbol K_i to denote the set of simplices in K of dimension i, for example $K_0 = X$.

Let us choose an ordering on the set X. For any n-dimensional simplex $\sigma = \{x_0 < x_1 < \cdots < x_n\} \in K_n$ and $0 \leq i \leq n$ define $d_i\sigma$ to be the $(n-1)$-dimensional simplex in K given by omitting x_i from σ.

A simplicial complex K on a finite set X can be realized as a topological space as follows: let us first choose linearly independent points $\{v_x\}_{x\in X}$ in some $\mathbb{R}^n$, then we define the realization of K to be the topological space given by the subset of the simplex $[v_x]_{x\in X}$ consisting of these points $\sum\limits_{x\in X} t_x v_x$ such that $\{x \in X \mid t_x \neq 0\}$ is a simplex in K.

Data points can be thought of as 0-simplices and one can build higher simplicial complexes on them such as the Vietoris-Rips complex or the witness complex. For computational details about constructing Rips or Witness complexes we refer the reader to (Singh, 2008).

A triangulation of a topological space X is a simplicial complex whose realization is either homeomorphic or homotopic to X; for example the sphere can be triangulated by the surface of a 3-simplex i.e. a tetrahedron, see Figure 2.3.

2.4 Euler characteristic

Key topological and homotopy invariants of topological spaces are constructed from triangulations. For example, the Euler characteristic $\chi(K)$ of a simplicial complex K is given by the alternating sum $|K_0| - |K_1| + |K_2| - \cdots + (-1)^i|K_i| + \cdots, 0 \leq i \leq k$. This is an example

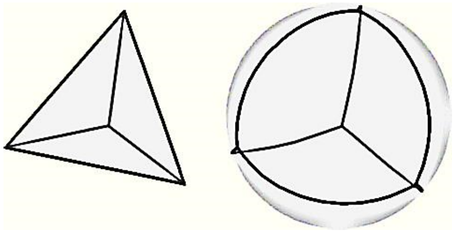

FIGURE 2.3: Triangulation of a sphere using a tetrahedron.

of a topological invariant since for a space X any of its triangulations have the same Euler characteristic. Furthermore, the Euler characteristic is homotopy invariant as any two homotopy equivalent spaces have the same Euler characteristic.

2.5 Homology

Let R be a commutative ring. A free R-module on a set S is a direct sum $\oplus_{s \in S} R$ of copies of R. Let X be a set and K be a simplicial complex on X. Define $C_n(K) := \oplus_{\sigma \in K_n} R$.

Then choose an ordering of X and define the following sequence of homeomorphisms:

$$\cdots \xrightarrow{\partial_{n+1}} C_n(K) \xrightarrow{\partial_n} C_{n-1}(K) \to \cdots \to C_1(K) \xrightarrow{\partial_1} C_0(K) \to 0$$

where ∂_n sends the generator e_σ of the components of R in $\oplus_{\sigma \in K_n} R$ indexed by $\sigma \in K_n$ to the alternating sum of generators $e_{d_0\sigma} - e_{d_1\sigma} + \cdots + (-1)^n e_{d_n\sigma}$ (see section 2.3 for the definition of $d_i\sigma$). It is standrad to see that the composition $\partial_n \circ \partial_{n+1}$ is trivial, see for example (Mac Lane, 1963). This means that the above sequence is a chain complex and Im $\partial_{n+1} \subset$ Ker ∂_n. The elements of Im ∂_{n+1} are called the boundaries and the elements of Ker ∂_n are called cycles. For example, the boundary $\partial[v_1, v_2]$ of an edge is given by $v_2 - v_1$. Also the boundary of a tetrahedron is its four triangle faces. We can take the boundary of a cycle and find out that it is equal to zero since:

$$\partial\left([v_1, v_2] + [v_2, v_3] - [v_1, v_3]\right) = (v_2 - v_1) + (v_3 - v_2) - (v_3 - v_1) = 0$$

2.5.1. The quotient Ker ∂_n/Im ∂_{n+1} is called the homology group of K with coefficients in R and is denoted by $H_n(K, R)$.

An important observation is that the isomorphism type of the R-module $H_n(K, R)$ does not depend on the ordering we have chosen on X. For our purposes we are mainly interested in the case R being a finite field $\mathbb{F}_p$.

2.5.2. Consider a field R. The dimension of the R-vector space denoted by $H_n(K, R)$ is called the n-th Betti number of K with respect to R and is denoted by $\beta_n(K, R)$. The 0-th Betti number $\beta_0(K, R)$ coincide with the number of connected components, hence it is field independent. It is important to realize that higher Betti numbers in general do depend on the field chosen. However for spaces like circle, sphere or torus, the Betti numbers are field independent.

Recall that the Euler Characteristic can also be calculated as an alternating sum:

$$\chi(K) = \beta_0(K, R) - \beta_1(K, R) + \cdots + (-1)^i \beta_i(K, R) + \cdots$$

Thus although individual Betti numbers $\beta_i(K, R)$ are field dependent, their alternating sum is not.

Here are some examples of Betti numbers (in those cases shown we do not need to specify a field as the Betti numbers here are field independent):

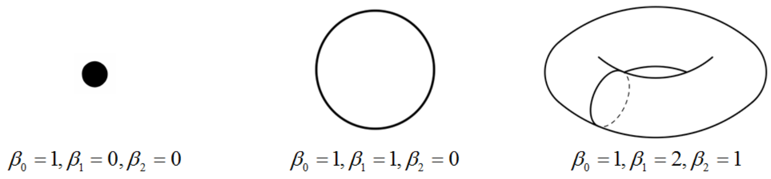
$\beta_0 = 1, \beta_1 = 0, \beta_2 = 0$
$\beta_0 = 1, \beta_1 = 1, \beta_2 = 0$
$\beta_0 = 1, \beta_1 = 2, \beta_2 = 1$

FIGURE 2.4: The first three Betti numbers for a point, a circle and a torus.

In summary, Betti numbers are designed to measure the number of connected components, cycles, voids and their higher dimensional analogues. They are invariants that can effectively be calculated and likewise are convenient when distinguishing between spaces. Effectiveness comes from the fact that one needs only to use basic linear algebra tools to calculate those numbers.

2.6 Persistent homology and data

Let R be a field. The homology $H_n(K, R)$ is an algebraic representation of certain topological features that X possess. Its dimension (which is called Betti numbers) represents a count of those features. For example $\beta_0(K, R)$ is the number of connected components of K, $\beta_1(K, R)$ is the number of certain loops and $\beta_2(K, R)$ of certain voids.

Now in the world of data we unfortunately do not have a direct access to a topological space or to a simplicial complex; instead we have access to its sampling and persistence is going to help us extend homology to the world of data and remarkably allows us to recover homological features such as holes in the space out of the sampling.

The basic approach of persistent homology is to sum up balls of size $\varepsilon > 0$ around each data point of the sampling, then calculate the homology of the obtained space or simplicial complex and finally record how this homology is changing while varying $\varepsilon > 0$.

Now we mention three constructions used to build relevant simplicial complexes out of a point cloud. A point cloud for us is simply a finite set X. A starting point is to choose a metric d on X. This choice is often suggested by the experiment that produces the data and likewise what we are interested in measuring or understanding about the data. Typically there is no one particular good choice of a metric and it is often advisable to perform the analysis with respect to different metrics and compare the results. An important aspect of topological data analysis is that the outcome is not sensitive to small changes in metrics.

2.6.1. Vietoris-Rips complex. Let $\varepsilon > 0$ be a real number. Define $V(X, \varepsilon)$ to be a simplicial complex on the set X consisting of these subsets $\sigma \subset X$ where $d(x, y) < \epsilon$ for any $x, y \in \sigma$. Note that $V(X, \varepsilon) \subseteq V(X, \varepsilon')$ for $\varepsilon \leq \varepsilon'$.

2.6.2. Čech complex. Let $\varepsilon > 0$ be a real number. Define $C(X, \varepsilon)$ to be a simplicial complex on the set X consisting of these subsets $\sigma \subset X$ for which there is $y \in X$ such that $d(x, y) < \epsilon$ for any $x \in \sigma$. Note that in this case as well we get $C(X, \varepsilon) \subseteq C(X, \varepsilon')$ for $\varepsilon \leq \varepsilon'$.

This is an example of a more general nerve complex of a covering of a topological space (T. K. Dey and Wang, 2015). Given a finite covering $\mathbb{U} = \{U_\alpha\}_{\alpha \in A}$ of a topological space, its nerve is the simplicial complex $N(\mathbb{U})$ on the set A, and where a subset $\{\alpha_0, \alpha_1, \ldots, \alpha_k\} \subseteq A$ is a k-simplex in $N(\mathbb{U})$ if and only if $\mathbb{U}_{\alpha_0} \cap \mathbb{U}_{\alpha_1} \cap \ldots \cap \mathbb{U}_{\alpha_k} \neq \emptyset$. For example, the Čech complex $C(X, \varepsilon)$ is the nerve complex of the covering $\{B(x, \epsilon)\}_{x \in X}$ of X where $B(x, \epsilon) = \{y \in X \mid d(x, y) < \epsilon\}$.

2.6.3. Witness vesrsions. Let $\varepsilon > 0$ be a real number. To construct these complexes we need to choose a subset $X_w \subset X$ of witnesses of X. Define $V_w(X, \varepsilon)$, the witness version of the Vietoris-Rips complex, to be a complex on X_w that consists of these $\sigma \subset X_w$ where for any $x, y \in \sigma$, there is $z \in X$ such that $d(x, z) < \epsilon$ and $d(y, z) < \epsilon$.

Similarly define $C_w(X, \varepsilon)$, the witness version of the Čech complex, to be a complex on X_w that consists of these $\sigma \subset X_w$ where there is $z \in X$ such that $d(x, z) < \epsilon$ for any $x \in \sigma$.

In any of these cases we have inclusions for any $\varepsilon \leq \varepsilon'$ such that

$$V_w(X, \varepsilon) \subseteq V_w(X, \varepsilon'), \qquad C_w(X, \varepsilon) \subseteq C_w(X, \varepsilon').$$

Witness complexes are very convenient to save both computational time and memory in particular when X is very large.

The outcome of the above constructions (Vietoris-Rips, Čech and their witness versions) is a sequence of complexes X_ϵ on a point cloud X, for any positive real number ϵ together with the inclusions $X_\epsilon \subset X'_\epsilon$ when $\varepsilon \leq \varepsilon'$.

Let R be a field. By choosing an ordering on X and applying homology to the outcome of our complex construction we get a sequence of vector spaces and induced maps for any increasing sequence of real numbers $\varepsilon_1 < \varepsilon_2 < \cdots < \varepsilon_i < \cdots$ according to

$$H_n(X_{\varepsilon_1}, R) \to H_n(X_{\varepsilon_2}, R) \to \cdots \to H_n(X_{\varepsilon_i}, R) \to \cdots$$

Such a sequence of vector spaces are called persistence diagrams or modules. Their key feature is that they can be characterized by discrete invariants called barcodes. The bars in a barcode represent the lifetime of features across filtration i.e. tracking when a feature is born and when it dies (P. Y. Lum, 2013). Using barcodes give us the possibility to measure the importance of homological features using their life spans. In this way we are able to get a meaning for what a statistically significant component such as a loop or a void is in a point cloud data.

In section 4 (Experiments with actual data) we are going to demonstrate some practical examples of computing persistent homology and their barcodes for some point clouds.

We finish this section with an example of Čech complex of a dataset given by a sample of points from a hemisphere. In this example we use the Euclidian metric. In Figure 2.5 we see that selecting the parameter ε to be too small will result in a descrete complex with n components, where n is the number of data point in the point cloud. On the other hand selecting ε to be too big will result in a full simplex on the whole dataset; a complex with trivial homology.

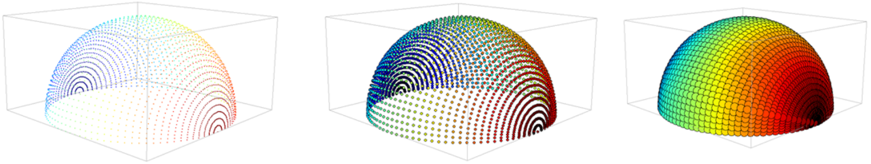

FIGURE 2.5: Growing points into balls of radius $\varepsilon > 0$ on a point cloud of a hemisphere.

2.7 Clustering algorithms

Let X be a topological space or a simplicial complex. Its connected components are also called clusters by computer scientists. The 0-th homology is a convenient tool to identify the components or the clusters of X. When dealing with real world data; instead of having a topological or a metric space, we are given its sampling. An important question is how to recover the clusters of the space out of its sampling.

A sampling is simply a finite metric space. An abstract finite metric space could be however a sampling of many spaces, with very different number of connected components. In many situations, in fact, we do not know which topological space we should choose which in return leads to different ways of clustering algorithms.

In general, the goal of a clustering algorithm is to partition observations into a certain number of clusters with similar attributes. This notion of similarity should reflect the topology of the space that the sampling was taken from. It is, however, important to note that one can choose different notions of similarity and each of those choices can lead to different clustering outcome. For example, if our data consists of gene expressions of certain number of individuals, one notion of similarity could be based on the expression similarity patterns; another notion of similarity could be the age of the individuals involved in the study. Those two notions of similarity often lead to different clustering outcomes and therefore there is no ultimately one good choice. Instead we can, in fact, use many different distances on our dataset; then try to compare different clustering outcomes in order to learn about the dataset. In any case the input for a clustering algorithm is a finite metric space X. There are several clustering algorithms that have been developed over the years. However, we are going to briefly mention three such methods and we refer the reader to (Carlsson and Mémoli, 2010; Müllner, 2013) for more details.

- *Single linkage:* we start with a matrix $\mathbf{D}$ of distances or similarities between N objects and merge the two nearest objects, say $C1$ and $C2$ corresponding to the smallest entry $\epsilon > 0$ of the matrix $\mathbf{D}$ to form a new cluster $[C1C2]$. Then the distances between the new cluster $[C1C2]$ and say, $C3$ is computed by $d_{[C1C2]C3} = min(d_{[C1C3]}, d_{[C2C3]})$.Finally, the matrix $\mathbf{D}$ is updated with the new distances and the procedure is repeated. The result can be graphically represented in the form of a dendrogram.

- *Complete linkage:* we start similarly with a matrix $\mathbf{D}$ of distances or similarities between N objects and merge the nearest two objects to form a new cluster $[C1C2]$. However, the distances between the new cluster $[C1C2]$ and $C3$ is computed by $d_{[C1C2]C3} = max(d_{[C1C3]}, d_{[C2C3]})$.

- *Average linkage:* Here the average inter-cluster distance between $[C1C2]$ and $C3$ is computed by $d_{[C1C2]C3} = \frac{d_{[C1]C3}+d_{[C2]C3}}{2}$. Then the matrix $\mathbf{D}$ is updated with the new distances and the procedure is repeated.

Among the above clustering schemes, the single linkage is the continuous one (Carlsson and Mémoli, 2010). That is because it can be recovered in the following way: for a finite metric space X and

$\epsilon > 0$, construct the Čech complex $C(X, \varepsilon)$; the set of connected components of this complex is in bijection with the set of clusters of the single linkage algorithm for that ϵ. This means that the clusters of the single linkage algorithm correspond to components of a space.

2.8 The Mapper algorithm

Mapper is a standard algorithm used to summarize vital information about point cloud datasets based on the idea of partial clustering and the use of filter functions defined on the data at hand. The output of Mapper is a graph.

The Mapper algorithm was introduced by Singh, Mémoli and Carlsson as a geometrical tool to analyze and visualize datasets (G. Singh and Carlsson, 1991). The idea behind Mapper is illustrated in Figure 2.6 and can be presented as follows: suppose we have a point cloud data representing a shape, for example a circle. We color the circle by filter values and project on a coordinate in order to reduce complexity via dimensionality reduction. Now the point cloud data is covered with overlapping intervals and therefore it is broken into overlapping bins. Afterwards, we collapse the points in each bin into clusters using a clustering algorithm. Once clustering is done, we can then create a network where each cluster of a bin is represented by a vertex and we can draw an edge when there is non-empty intersection between clusters.

The network can likewise be called a topological summary of the space data is sampled from. We refer the reader to (Singh, 2008), (P. Y. Lum, 2013) and (Carlsson, 2009) for more details.

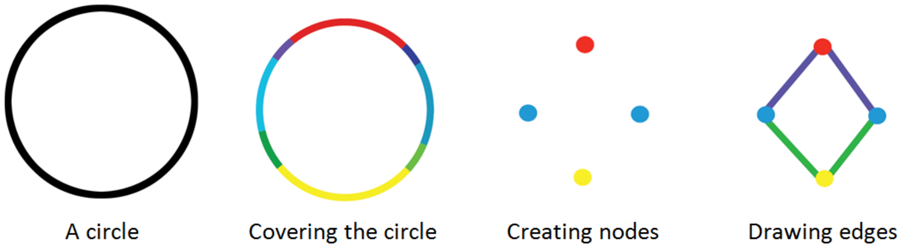

FIGURE 2.6: The mapper approach is applied to a circle; first the circle is covered by the values of the chosen filter function, then the range of those filter values are formed into overlapping intervals and finally nodes are created and an edge is drawn between two nodes if they have at least one element in common (Carlsson, 2009).

Definition 2.8.1 (Mapper, (T. K. Dey and Wang, 2015))**.** Let X and Z be topological spaces and let $f : X \to Z$ be a continuous map. Let $\mathbb{U} = \{U_\alpha\}_{\alpha \in A}$ be a finite open covering of Z. Consider the cover $f^* (\mathbb{U}) = \{f^{-1}(U_\alpha)\}_{\alpha \in A}$ of X, which is called the pull-back of $\mathbb{U}$ along f. The mapper construction arising from these data is defined to be the nerve complex (see 2.6.2) of the pullback cover $M(\mathbb{U}, f) := N(f^*(\mathbb{U}))$.

To use mapper we need the following input:

- a metric space X,
- a filter function $f : X \to Z$ (often Z is taken to be $\mathbb{R}$),
- a covering $\mathbb{U}$ of Z (a standard choice for the cover in the case $Z = \mathbb{R}$ is a finite sequence of interleaving intervals).

For this thesis we use the python Mapper software implementation of the mapper algorithm (A. Babu, 2013). As mentioned above, parameters such as choice of metric, filter function, type of cover or clustering algorithm needs to be adjusted.

A particularly important parameter to choose is the filter function. In the following section we briefly discuss some possible choices.

2.9 Filter functions

We are most interested in filter functions of the form $f : X \to \mathbb{R}$. In fact one can generalize filter functions, for example to the two dimensional plane or the three dimensional space, however, the important thing is to cover the dataset with overlapping sets; it could be for example, overlapping rectangles instead of intervals or any other shape depending on the problem at hand. The choice of a filter functions is very important since a good function can help us to reveal some interesting geometrical information about the dataset. There are several filter functions such as the kernel density estimator, distance to measure, eccentricity or principal metric SVD filters (P. Y. Lum, 2013) and (A. Babu, 2013).

Given a matrix of data points one can apply singular value decomposition in order to obtain the k-th eigenvector of a distance matrix, for example the principal eigenvector corresponds to the largest eigenvalue in magnitude. Projecting data points onto, for example, the principal eigenvector is a way for achieving dimensionality reduction; this projection can serve as a filter function and we can therefore produce a topological summary. Another projection yields a different filter function and therefore possibly a different-looking topological summary compared to the previous one.

Theorem 2.9.1 (Singular value decomposition (SVD),(Demmel., 1997)). *Let A be an arbitrary m-by-n matrix with $m \geq n$. Then we can write $A = U\Sigma V^T$, where U is m-by-n and satisfies $U^TU = I$, V is n-by-n and satisfies $V^TV = I$, and $\Sigma = diag\,(\sigma_1, \ldots, \sigma_n)$, where $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$. The columns $u_1, \ldots, u_n$ of U are called left singular vectors. The columns $v_1, \ldots, v_n$ of V are called right singular vectors. The σ_i are called singular values. (If $m < n$, the SVD is defined by considering A^T).*

The standard singular value decomposition is a very useful technique and in fact it is used for many purposes other than being a very valuable filter function.

3. Statistical learning

The outline of this part is as follows; first we start by reviewing some useful results from the theory of probability since probability theory provides the foundation upon which mathematical statistics relies on. Then we summarize the mathematical development of the statistical methods used in this thesis such as regression models and regularization as well as validation methods.

Statistical learning is concerned with understanding data through a vast set of supervised and unsupervised statistical tools. In general supervised learning has the goal of predicting the value of a response variable based on a set of predictors while in unsupervised learning there is no response and the objective is to describe relations or associations between a set of predictors.

3.1 Some basic definitions from probability theory

In probability theory one studies mathematical models of random phenomena that are intended to describe random experiments. A random experiment is as an experiment whose outcome cannot be predicted with certainty, for instance having the knowledge that a coin has a symmetric construction still cannot give us the power to predict the outcome of future tosses with certainty. For a random experiment the set Ω of all possible outcomes ω is usually called the outcome space, for example Ω can be the set of elements of subintervals of the real line $\mathbb{R}$ or it can be the two outcomes resulting from a coin tossing experiment; namely heads or tails $\Omega = \{H, T\}$.

We review some basic definitions and results:

3.1.1 (($\sigma-$algebra, (Borovkov, 2013)). A class of sets $\mathcal{F}$ of a sample space Ω is called $\sigma-$ algebra if the following conditions are met:

1. The empty set $\emptyset \in \mathcal{F}$.
2. If a collection of possible outcomes $A \in \mathcal{F}$, then the complement $A^{\complement} \in \mathcal{F}$.

3. If $\{A_n\}$ is a sequence of sets from $\mathcal{F}$, then their union $\bigcup\limits_{n=1}^{\infty} A_n \in \mathcal{F}$ and $\bigcap\limits_{n=1}^{\infty} A_n \in \mathcal{F}$.

If Ω is countable then one may consider the trivial $\sigma-$algebra that contains all the subsets of Ω i.e. the power set, for example the set of all events on $\Omega = \{H, T\}$ is the collection of events $\mathcal{F} = \{\varnothing, \{T, T\}, \{H, H\}, \{H, T\}, \{T, H\}, \Omega\}$

By contrast if Ω is uncountable, for example when $\Omega = \mathbb{R}$ then it is not possible to find a measure on all subsets of Ω. Therefore $\sigma-$algebra was introduced since a $\sigma-$ algebra can be smaller than the power set which contains too many subsets of Ω and we often are interested in considering only some subsets.

A pair $(\Omega, \mathcal{F})$ consisting of the outcome space Ω and a $\sigma-$algebra $\mathcal{F}$ is called a measurable space and the elements of $\mathcal{F}$ are known as measurable sets or events. One can for instance perform some elementary set operations on two events A and B of $\mathcal{F}$; for example to find their union $A \cup B = \{x : x \in A \vee x \in B\}$, their intersection $A \cap B = \{x : x \in A \wedge x \in B\}$ or the complement of one of them $A^{\complement} = \{x : x \notin A\}$.

On the real line $\mathbb{R}$ there is a special $\sigma-$algebra known as the Borel $\sigma-$algebra denoted by $\mathcal{B}$ and its elements are called Borel sets and the measurable space is the pair $(\mathbb{R}, \mathcal{B})$.

3.1.2 (Probability triple $(\Omega, \mathcal{F}, P)$, (Gut, 2012))**.** The triple $(\Omega, \mathcal{F}, P)$ is a probability (measure) space if:

- Ω is the sample space; that is some (possibly abstract) set.
- $\mathcal{F}$ is a $\sigma-$algebra of sets (events) - the measurable subsets of Ω.
- P is a probability measure.

The probability measure P satisfies the following three Kolmogorov axioms:

1. For any $A \in \mathcal{F}$, there exist a number $P(A)$ known as the probability of A and satisfying $P(A) \geq 0$.
2. $P(\Omega) = 1$.
3. If $A_1, A_2, \ldots \in \mathcal{F}$ are pairwise disjoint events, $A_k \cap A_l = \emptyset$ for $k \neq l$, then $P\left(\bigcup\limits_{i=1}^{\infty} A_i\right) = \sum\limits_{i=1}^{\infty} P(A_i)$.

3.2 Independence and conditional probabilities

Independence means that successive experiments do not influence each other. Two events A and B are independent if and only if the probability of their intersection equals the product of their individual probabilities i.e. $P(A \cap B) = P(A) \cdot P(B)$. The same concept can be extended to a collection of events $\{A_k, 1 \leq k \leq n\}$ such that they are jointly independent if and only if for any sub-collection of $\{1, 2, \ldots, n\}$ the condition $P\left(\bigcap A_{i_k}\right) = \prod P\left(A_{i_k}\right)$ is met. Hence pairwise independence is not sufficient for joint independence.

3.2.1 (Conditional probability, (Borovkov, 2013))**.** Let $(\Omega, \mathcal{F}, P)$ be a probability space and two events A and B are in Ω. If $P(B) > 0$, then the conditional probability A given B has occurred is defined by $P(A|B) := \frac{P(A \bigcap B)}{P(B)}$. If in particular A and B are independent, then $P(A|B) = \frac{P(A \bigcap B)}{P(B)} = \frac{P(A) \cdot P(B)}{P(B)} = P(A)$.

Two events A and B are conditionally independent given an event C with $P(C) > 0$ if $P(A \cap B|C) = P(A|C)P(B|C)$. An illustrative example is as follows; it might seem that there exists a correlation between the numbers of stork sightings and new-born babies. Both events in fact depend on common environmental factors and therefore both events conditionally on those environmental factors are independent (Husmeier, 2006).

Next if we partition the sample space Ω into a collection of disjoint events such that $\Omega = \bigcup\limits_{i=1}^{n} A_i$, where $A_i \cap A_j = \emptyset$ for $1 \leq i,\ j \leq n,\ i \neq j$, then for any event $B \subset \Omega$ the law of total probability is given by $P(B) = \sum\limits_{k=1}^{n} P(B|A_i) \cdot P(A_i)$.

Furthermore for any event $B \subset \Omega$ such that $P(B) > 0$, the Bayes formula is given by $P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum\limits_{j=1}^{n} P(B|A_j)P(A_j)}$.

3.3 Random variables and random vectors

We consider a random variable X to be a set function from a probability space $(\Omega, \mathcal{F}, P)$ to the real line $\mathbb{R}$ i.e. $X : \Omega \to \mathbb{R}$.

3.3.1 (Random variable, (Borovkov, 2013))**.** A random variable X is a measurable function $X = X(\omega)$ mapping $(\Omega, \mathcal{F})$ into $(\mathbb{R}, \mathcal{B})$, where $\mathbb{R}$ is the set of real numbers and $\mathcal{B}$ is the $\sigma-$algebra of all Borel sets, i.e. a function for which the inverse image of any Borel set is $\mathcal{F}$-measurable $X^{-1}(B) = \{\omega : X(\omega) \in B\} \in \mathcal{F}, \forall B \in \mathcal{B}$.

In probability theory a measurable function can be denoted by X. If X is measureable from $(\Omega, \mathcal{F})$ to $(\mathbb{R}, \mathcal{B})$, then X is a random variable iff $\{X \leq x\} = \{\omega \in \Omega : X(\omega) \leq x\} \in \mathcal{F}, \forall x \in \mathbb{R}$.

To each random variable we can associate a probability measure such that $P(A) = P\left(X^{-1}(A)\right) = P\left(\{\omega : X(\omega) \in A\}\right), \forall A \in \mathcal{B}$.

A very useful example of a measurable function is the indicator function for the measurable set A, $A \subset \Omega$, $I_A : \Omega \to \{0, 1\}$ defined as

$$I_A(\omega) = \begin{cases} 1, & \omega \in A \\ 0, & \omega \notin A \end{cases}$$

A more general case is when there are multiple random variables, for example a set of several features can be denoted as an n-dimensional random vector $\mathbf{X} = (X_1, X_2, \ldots, X_n)'$, where $(\boldsymbol{.})'$ denotes the transpose and n is the number of elements in $\mathbf{X}$.

A random vector $\mathbf{X}$ is a measurable function from Ω to $\mathbb{R}^n$ i.e. $\mathbf{X} : \Omega \to \mathbb{R}^n$ and we think of $\mathbf{x} = (x_1, x_2, \ldots, x_n)'$ as the outcome of the vector $\mathbf{X}$. Similarly the inverse image of any Borel set is $\mathcal{F}-$ measurable $\mathbf{X}^{-1}(B) = \{\omega : \mathbf{X}(\omega) \in B\} \in \mathcal{F}, \forall B \in \mathcal{B}^n$.

3.4 Distribution functions

One can associate to every random variable X a function called the cumulative distribution function of X denoted by $F_X(x)$ and defined as $F_X(x) = P(X \le x) = \int\limits_{-\infty}^{x} f_X(u)\,du, \forall x \in \mathbb{R}$.

The function $f_X(x)$ is called the probability density function of X such that $\int\limits_{-\infty}^{\infty} f_X(x)\,dx = 1, f_X(x) \ge 0, \forall x \in \mathbb{R}$ and the function $p_X(x_k)$ is called the probability mass function of X such that $\sum\limits_{k=-\infty}^{\infty} p_X(x_k) = 1, p_X(x_k) \ge 0$.

The joint distribution function of a random vector $\mathbf{X}$ is $F_{\mathbf{X}}(\mathbf{x}) = P(\mathbf{X} \le \mathbf{x}) = P(X_1 \le x_1, X_2 \le x_2, \ldots, X_n \le x_n), \mathbf{x} \in \mathbb{R}^n$. The joint density function of a random vector $\mathbf{X}$ or the multivariate probability density function $f_{\mathbf{X}}(\mathbf{x})$ of $\mathbf{X}$ is defined as the derivative of the cumulative distribution function $F_{\mathbf{X}}(\mathbf{x})$ with respect to the component of $\mathbf{X}$ i.e. $f_{\mathbf{X}}(\mathbf{x}) = \frac{\partial^n}{\partial x_1 \partial x_2 \cdots \partial x_n} F_{X_1, X_2, \ldots, X_n}(x_1, x_2, \ldots, x_n)$.

A useful example is the Bernoulli distribution. Consider an experiment such as coin tossing conducted with its outcomes determined as either heads or tails and let the probability of heads to be denoted by p and consequently the probability of tails to be $1-p$. The Bernoulli distribution is defined as $f(x) = p^x (1-p)^{1-x}$, for $x = 0, 1$. The probability that k successive tosses result in heads is given by the probability mass function $p_X(x) = P(X = x) = \binom{n}{x} p^k (1-p)^{n-k}$.

Important continuous distributions are the normal distribution and the F-distribution. A random variable X has a normal distribution or Gaussian distribution $X \in N(\mu, \sigma^2)$ if it has range $(-\infty, \infty)$ and a probability density function $f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$. The parameters of the distribution are the mean μ and the variance σ^2, where $E\left[X\right] = \mu$ and $Var\left[X\right] = \sigma^2$. The standard normal distribution denoted by $N\left(0,1\right)$ is obtain when $\mu = 0$ and $\sigma^2 = 1$. The $k-$dimensional normal density for the random vector $\mathbf{X}$ has the form $f_{\mathbf{X}}\left(\mathbf{x}\right) = \frac{1}{(2\pi)^{k/2}\sqrt{\det(\mathbf{C}_{\mathbf{X}})}} \exp\left(-(\mathbf{x}-\mu)'\mathbf{C}_{\mathbf{X}}^{-1}\left(\mathbf{x}-\mu\right)/2\right)$.

Normally distributed random variables are independent if they are uncorrelated. The coefficient of correlation ρ of X and Y is defined as $\rho_{X,Y} = \frac{Cov(X,Y)}{\sqrt{Var(X)\cdot Var(Y)}}$. The coefficient $\rho_{X,Y}$ is a measure of the relationship between X and Y. The value is always between -1 and 1. If $\rho_{X,Y}$ is positive (or negative), then X and Y is positively (or negatively) related and when $\rho_{X,Y} = 0$ then X and Y are said to be uncorrelated.

If X and Y are independent, then they are uncorrelated. However the contrary is not necessarily true since uncorrelated random variables may not be pairwise independent.

3.5 Expectation and conditional expectation

The expectation and the second central moment if they exist are two important characteristics of a random variable and its distribution. Also Conditional probabilities and conditional expectation are fundamental in probability theory since the expectation of a random variable usually depends on information, for example life expectancy may depend on location, diet, gender or other information.

3.5.1 (Expectation of a function of a random variable, (Kroese and Chan, 2014))**.** The expectation of X if it exists can be computed as

$$E\left[g\left(X\right)\right] = \begin{cases} \sum\limits_{k=-\infty}^{\infty} g\left(x_k\right)p_X\left(x_k\right) & \text{discrete r.v} \\ \int_{-\infty}^{\infty} g\left(x\right)f_X\left(x\right)dx & \text{continuous r.v} \end{cases}$$

The expectation of the indicator function I_A is defined as $E\left[I_A\right] = P\left(A\right)$.

More general let X be a random variable and g a Borel function such that $E\left[g\left(X\right)\right] < \infty$, the law of the unconscious statistician states that $E\left[g\left(X\right)\right] = \int_{\Omega} g\left(X\right)dP = \int_{-\infty}^{\infty} g\left(x\right)dF\left(x\right)$.

3.5.2 (Conditional expectation, (Borovkov, 2013))**.** Let X be a random variable on a probability space $\left(\Omega, \mathcal{F}, P\right)$ where $E\left[|X|\right] < \infty$ and take $A \in \mathcal{F}$ such that $P\left(A\right) > 0$, then the conditional expectation of X given A is defined as $E\left[X|\, A\right] = \frac{1}{P(A)}\int_A XdP$.

One can compute the conditional expectation of X given the partitioning sets $A_i \in \mathcal{F},\ i = 1,2,\ldots,k,\ A_i \bigcap A_j = \varnothing,\ j \neq i,\ \bigcup\limits_{i=1}^{k} A_i = \Omega$ using $E\left[X|\, A_i\right] = \frac{1}{P(A_i)}\int_{A_i} XdP$.

An application of the properties of conditional expectation with respect to a $\sigma-$algebra is that an estimator of Y based on X can be expressed as $\widehat{Y} = E\left[Y|\,\mathcal{F}_X\right] = E\left[Y|\,X\right]$ with the estimation error $\widetilde{Y} = Y - \widehat{Y}$, where Y is a random variable such that $Var\left[Y\right] < \infty$ and X is another random variable in the same probability space.

3.6 Regression models

Suppose we are interested in investigating the linear relationship between a random variable Y whose mean is depending on a numeric non-random variable x_1. This relationship can be expressed as $\mu(x_1) = E\left[Y|x_{\underline{1}}\right] = \beta_0 + \beta_1 x_1$ or equivalently to $Y = \mu(x_1) + \varepsilon = \beta_0 + \beta_1 x_1 + \varepsilon, E\left[\varepsilon\right] = 0$.

The multiple regression setting is written as $y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \varepsilon_i, i = 1, 2, \ldots, n$. The regression model is used to describe the relationship between responses or dependent variables y_i and a set of observational predictors or independent variables $x_{i1}, x_{i2}, \ldots, x_{ik}$.

The linear model can be generalized to a non-linear model, however the linear model has advantages in terms of inference and sometimes accuracy. In matrix notation the general linear model has the form

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots x_{1k} \\ 1 & x_{21} & x_{22} & \cdots x_{2k} \\ \vdots & \vdots & \vdots & \ddots \ \vdots \\ 1 & x_{n1} & x_{n2} & \cdots x_{nk} \end{pmatrix}, \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}, \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

The vector $\mathbf{y}$ of dimension $n \times 1$ is the observed responses, the $n \times (k+1)$ matrix $\mathbf{X}$ whose rows holds the values of variables associated with each observation y_i, the vector β of dimension $(k+1) \times 1$ is the unknown parameters β_i and finally ε is a $n \times 1$ vector of random errors that are assumed to be uncorrelated with mean zero and constant variance σ^2 i.e. $E\left[\varepsilon\right] = \mathbf{0}, \mathrm{cov}\left(\varepsilon\right) = \sigma^2 \mathbf{I}_n$.

The error terms are independently distributed normal random variables $\varepsilon_i \in N\left(0, \sigma^2\right)$. One typically fits this model using the unbiased least squares coefficient estimator that minimizes the quantity $\sum\limits_{i=1}^{n} \left(y_i - \left(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik}\right)\right)^2$. The estimated expected response value given the set of observed predictors is called the fitted value and is equal to $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2} + \cdots + \hat{\beta}_k x_{ik}$. The difference between the observed value y_i and the fitted value $\hat{y}_i$ is the residual $\hat{\varepsilon}_i$.

The value $\hat{\beta}$ is the OLS (Ordinary Least Squares) of β that minimizes the sum of the squares $\hat{\varepsilon}^t\hat{\varepsilon} = \|\hat{\varepsilon}\|^2$ of the residuals. The OLS estimate of β is $\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$.

An unbiased estimator does not systematically over- or underestimate the true parameter; therefore over a large number of trials one would expect the error to sum to approximately zero. The value $\hat{\beta}$ is the unbiased estimate of β since $\hat{\beta} - \beta = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon$ and therefore $E\left[\hat{\beta}\right] = \beta$.

The residual sum of squares (RSS) is $\sum\limits_{i=1}^{n}(y_i - \hat{y}_i)^2 = \hat{\varepsilon}_1 + \hat{\varepsilon}_2 + \cdots + \hat{\varepsilon}_n = \|\hat{\varepsilon}\|^2$. A measure of goodness of fit is the R^2 statistic defined as $R^2 = corr(y_i, \hat{y}_i)^2 = \rho(y_i, \hat{y}_i)^2$. An R^2 value close to one indicates that a large proportion of the variability in the response has been explained by the regression. A value near zero indicates that the regression did not explain much of the variability in the response.

A measure of the lack of fit of the model to the data is called the residual standard error (RSE): $s = \sqrt{\frac{1}{n-k-1}RSS} = \sqrt{\frac{1}{n-k-1}\|\hat{\varepsilon}\|^2}$. Standard errors (SE) can be used to perform hypothesis testing on the hypothesis coefficients

$$\begin{cases} H_0 : \beta_j = 0 \\ H_0 : \beta_j \neq 0 \end{cases}.$$

In order to test the null hypothesis, one needs to determine whether $\hat{\beta}_j$ is sufficiently far from zero; thus we can be confident that β_j is non-zero. A small p-value indicates that it is unlikely to observe such a substantial association between the predictor and the response due to chance. One rejects the null hypothesis if the p-value is small enough. Typically the p-value cut-offs for rejecting the null hypotheses are $\alpha = 5\%$ or 1%.

However when dealing with a large number of predictors for example 100 variables, we expect at the level $\alpha = 5\%$ to see five small p-values even in the absence of true associations just due to chance which may lead to false findings. Moreover the R^2 value will always increase when more variables are added to the model even though those variables are weakly associated with the response.

3.7 Logistic regression

Logistic regression is a widely-used probability model when the response is binary valued i.e. equal only to 1 or 0. The logistic regression model can be expressed as

$$E\left[Y\,|x\right] = P\left\{Y = 1\,|x\right\} = \frac{\exp\left(\beta_0 + \beta_1 x\right)}{1 + \exp\left(\beta_0 + \beta_1 x\right)}$$

The logit transformation defined in terms of the conditional mean $\pi\left(x\right) := E\left[Y\left|x\right.\right]$ is $\log\left(\frac{\pi(x)}{1-\pi(x)}\right) = \beta_0 + \beta_1 x$. In this method we choose those values of β_0 and β_1 that minimize the sum of squared deviations of the observed values Y. We can write the general case for p explanatory variables as $\pi = \frac{\exp(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}{1+\exp(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}$ and $\operatorname{logit}\left(\pi\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$.

Maximum likelihood method can be used to estimate values for the unknown logistic regression parameters that maximize the probability of obtaining the observed set of data. The so called likelihood function expresses the probability of the observed data as a function of the unknown parameters. For more details we refer the reader to the references (Balakrishnan, 2014), (Johnson and Wichern, 2013) and (Hastie, 2009).

3.8 Interactions

Suppose that a model has two predictors x_1 and x_2; likewise suppose that the effect of one of them cannot be separated from the other. For instance the effect of x_1 on the response Y depends on x_2. Hence we would like to describe this interaction by constructing a new predictor $x_3 = x_1 x_2$. The model then becomes $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$. In general interpreting interactions among more than two variables can be extremely complex.

As we mentioned before the linear model has advantages in terms of inference. Hence improving the linear model is useful and is achieved by using an alternative fitting procedure rather than the least squares fitting or the maximum likelihood approach. In general, using alternative procedures can generate better prediction accuracy and model interpretability.

We therefore introduce an improved fitting procedure known as regularization instead of the least squares approach.

3.9 Linear model selection and regularization

An alternative approach is collectively termed shrinkage methods and they provide better model interpretability and often improved predication accuracy.

When the least squares estimates have low variance, the performance is good on test observations; however, in other situations when the number of predictors p is almost as large as the number of observations n, the least squares estimates will suffer from high variance leading to overfitting and therefore poor prediction accuracy.

In the case of $k > n$, the least squares estimates do not have a unique solution. Shrinkage or regularization has the effect of reducing variance of the coefficient estimates at the cost of a negligible increase in bias. It involves fitting a model with all the predictors, then the estimates coefficients are shrunk towards zero and some of them possibly are estimated to be exactly zero which makes variable selection possible. Well-known methods that use shrinkage penalty are ridge regression, the lasso and logistic regression.

The ridge coefficient estimates $\hat{\beta}_R^\lambda$ are the values that minimize the quantity $\sum_{i=1}^{n}(y_i-(\beta_0+\beta_1x_{i1}+\beta_2x_{i2}+\cdots+\beta_kx_{ik}))^2+\lambda\sum_{j=1}^{k}\beta_j^2 = RSS+\lambda\sum_{j=1}^{k}\beta_j^2$. The first term involves minimizing the RSS in order to fit the data well. The second term involves minimizing the shrinkage penalty $\lambda\sum_{j=1}^{k}\beta_i^2$ because when it is small, the regression coefficients will approach zero. Also we have a tuning parameter λ to be determined in order to control the impact of those two terms. Then one obtains a set of coefficient estimates $\hat{\beta}_R^\lambda$ for each value of λ. Now as λ increases the flexibility of the ridge regression fit decreases and consequently the variance is reduced at the cost of a slight increase in bias. For example, when $\lambda=0$ the ridge regression estimates $\hat{\beta}_R^\lambda$ are the same as the least squares estimates $\hat{\beta}$, where the variance is high but there is no bias; however as $\lambda\to\infty$, the shrinkage penalty $\|\beta\|_2=\sqrt{\sum_{j=1}^{k}\beta_j^2}$ has more impact in the sense that all the coefficient estimates are shrunken towards zero but none of them will be exactly zero unless $\lambda=\infty$. The ridge regression leads to a considerable decrease in variance and improvement in prediction accuracy with a very little increase in bias but will not exclude any of the predictors which can be viewed as a disadvantage. One typically uses the quantity $\left\|\hat{\beta}_R^\lambda\right\|_2/\left\|\hat{\beta}\right\|_2$ to describe the amount of shrinkage of the ridge coefficient estimates; for example a small value indicates that all coefficient estimates have been shrunken towards zero. This quantity ranges from one when $\lambda=0$ down to zero as $\lambda\to\infty$.

The lasso method, however, is able to shrink the coefficient estimates to be exactly zero and overcomes the disadvantage of the ridge regression and therefore performing variable selection is possible.

The lasso coefficient estimates $\hat{\beta}_L^\lambda$ are the values that minimize the quantity $\sum_{i=1}^{n}(y_i-(\beta_0+\beta_1x_{i1}+\beta_2x_{i2}+\cdots+\beta_kx_{ik}))^2+\lambda\sum_{j=1}^{k}\left|\beta_j\right| = RSS+\lambda\sum_{j=1}^{k}\left|\beta_j\right|$. The penalty is L_1 norm and by increasing λ the shrinking is such that $\beta_j=0$ for values of j belong to some set of variables. Similar to ridge regression, when $\lambda=0$ the lasso regression estimates $\hat{\beta}_L^\lambda$ are the same as the least squares estimates $\hat{\beta}$, where the variance is high but there is no bias; however as λ grows sufficiently large, the shrinkage penalty becomes $\|\beta\|_1=\sum_{j=1}^{k}|\beta_j|$ and all the coefficient estimates are shrunken to be exactly zero.

In general, the lasso will perform better than ridge regression in situations such as when the response variable is related to a small number of predictors with substantial coefficient estimates since the lasso implicitly assumes that the remaining set of coefficients are exactly equal to zero. On the other hand, ridge regression assumes that the response is related to a set of predictors with coefficient estimates nearly equal to zero.

There is also the elastic net penalty which is a combination of ridge and the lasso regression. The elastic net for logistic regression solves the optimization problem

$$\min_{(\beta_0,\beta)\in\mathbb{R}^{p+1}} -\left[\frac{1}{N}\sum_{i=1}^{N} y_i\left(\beta_0 + x_i^T\beta\right) - \log\left(1+e^{\left(\beta_0+x_i^T\beta\right)}\right)\right] + \lambda\left[(1-\alpha)\left\|\beta\right\|_2^2/2 \ + \alpha\left\|\beta\right\|_1\right]$$

where N is the number of observations, λ is the turning parameter and $\|.\|_p$ is the L_p norm. If $\alpha = 0$ then we are dealing with the case of ridge regression. If $\alpha = 1$ then the model reduces to the lasso model. The elastic net occur when $0 < \alpha < 1$.

3.10 Overfitting and cross validation

It is very important to be able to validate observations about the data at hand. A good choice of a model is one that can predict well on new unseen data. A model is considered to be overfitted if it performs well on a training dataset but its performance on previously unseen data is poor.

One mechanism that can help reduce the risk of overfitting is called the k-fold cross-validation; it is a method that can be used to determine which model is the best for a given situation. In general the idea of cross validation is to split the data into k-training samples used by the classification method and k-test samples used for calculating misclassification error.

When we later apply the penalized logistic regression method in the data experimenting chapter, we will show how cross-validation provides a way for us to select a tuning parameter λ with respect to the misclassification error which in return helps us to determine which model to pick from a variety of possible models. For more details two comprehensive references are (Hastie, 2009) and (Hastie, 2015).

4. Experiments with actual data

Point cloud data are given as a finite metric space (X, d) and n data points can be viewed as n 0-simplices.

We are going to summarize topological information of point clouds into barcodes and persistence diagrams using the theory of persistent homology. Moreover, we are going to use the Mapper tool to produce topological summaries of point clouds.

4.1 Persistent homology on shapes of known topology

We start by examining some datasets generated from samples of objects whose topology is known such as a circle or a torus. We plot 40 points randomly chosen with uniform distribution from a circle of radius one. We then compute barcode and persistence diagram of the Rips filtration constructed on that point cloud, see Figure 4.1.

For details about computations we refer the reader to (J. K. Brittany T. Fasy, Ulrich Bauer, and Reininghaus., 2015). The barcode in Figure 4.1 shows black bars, a red bar and some blue bars.

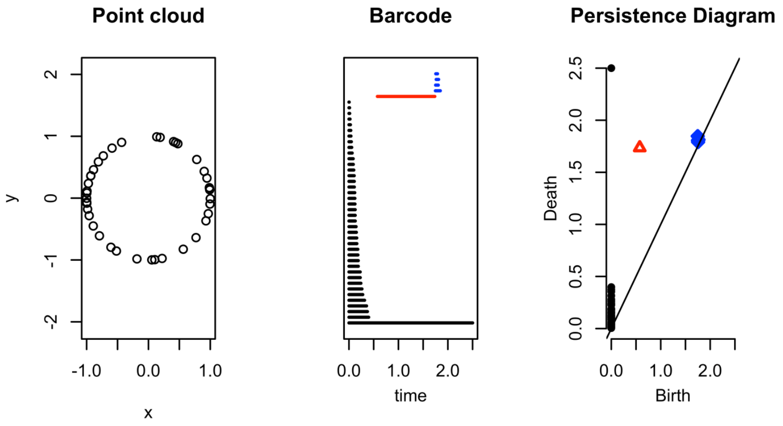

FIGURE 4.1: Barcode and persistence diagram of a point cloud of a sample of a circle.

First, the black bars represent the 0-dimensional cycles or vertices, i.e. cycles in H_0 and all those black bars start at time zero but can end at different times during the time evolution of the Rips filtration. Second, a red bar represents a one-dimensional cycle, i.e. a cycle in H_1. Third, the blue bars represent trapped voids, i.e. cycles in H_2. The persistence diagram in Figure 4.1 shows points representing birth and death of cycles in H_i. A black point is a cycle in H_0, a red triangle is a cycle in H_1 and a blue diamond is a cycle in H_2. Hence for the point cloud of a sample of a circle shown in Figure 4.1, we have one connected component and one loop.

Next in Figure 4.2 we generate a noisy circle using the previous circle but with added noise, namely adding a uniform random variable to each coordinate. The barcode still shows black bars and some red bars; however, one red bar seems to persist longer compared to the other red bars. Similarly the persistence diagram shows a black point that is distinguished from the rest of black points. It also shows one distinguished red triangle quite far from the diagonal which suggests that a cycle in H_1 is a significant feature. Notice that despite the noise added, we are still able to identify one connected component and one loop.

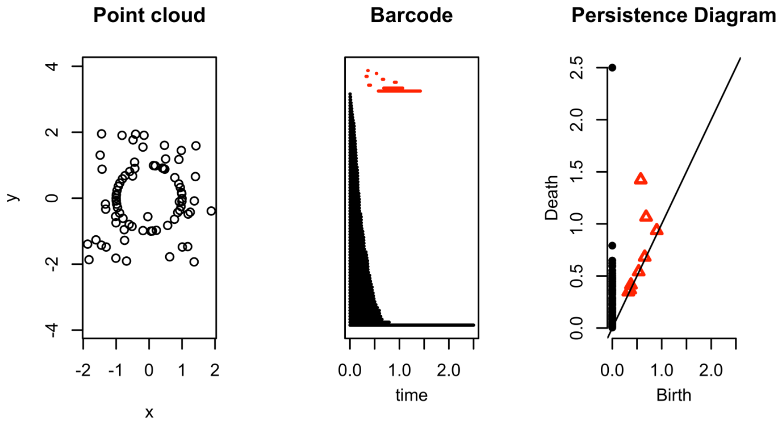

FIGURE 4.2: Barcode and persistence diagram of a point cloud of a sample of a noisy circle.

We continue experimenting and plot six data points in such a way that they are divided into three clusters and compute the number of connected components using the single linkage dendrogram. Hierarchical clustering in general fuses neighboring points to form larger clusters and it constructs a tree-like nested structure from a distance matrix.

In Figure 4.3 we see that both barcode and persistence diagram agree with the dendrogram showing three connected components that persisted for a relatively long filtration range.

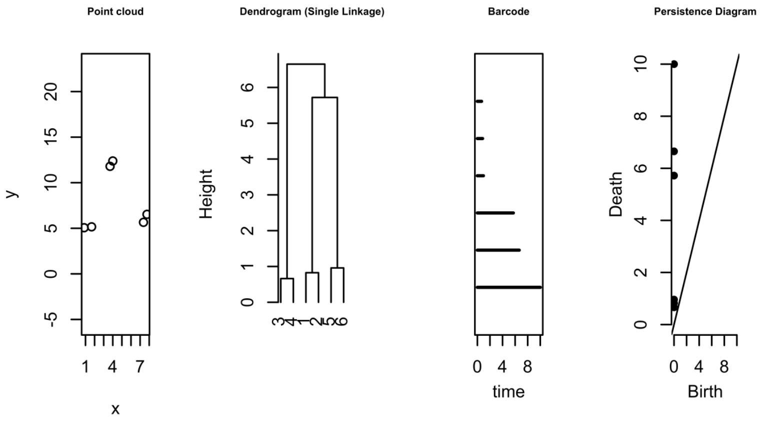

FIGURE 4.3: Barcode and persistence diagram of point cloud of 6 data points.

We can further create three dimensional shapes of known topology, for example a sphere sampled by selecting 300 points randomly from a sphere in $\mathbb{R}^3$ with radius equal to 1. Then we plot the barcode and the persistence diagram of the Rips filtration, see Figure 4.4.

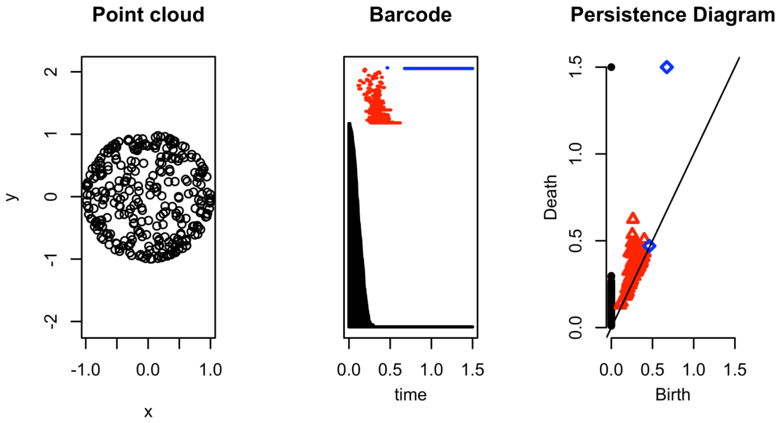

FIGURE 4.4: Barcode and persistence diagram of a point cloud of a sphere.

We can also create another three dimensional object, namely a torus sampled by selecting 300 points randomly from a torus in $\mathbb{R}^3$ where the radius of the torus tube is equal to one and the radius from the centre of the hole to the centre of the torus tube is equal to two as shown in Figure 4.5. Then we plot the barcode and the persistence diagram as mentioned before.

In Figure 4.6, we show a 16-by-16 pixel image of a handwritten digit eight; that is to say a vector in $\mathbb{R}^{256}$. Notice that the barcode and

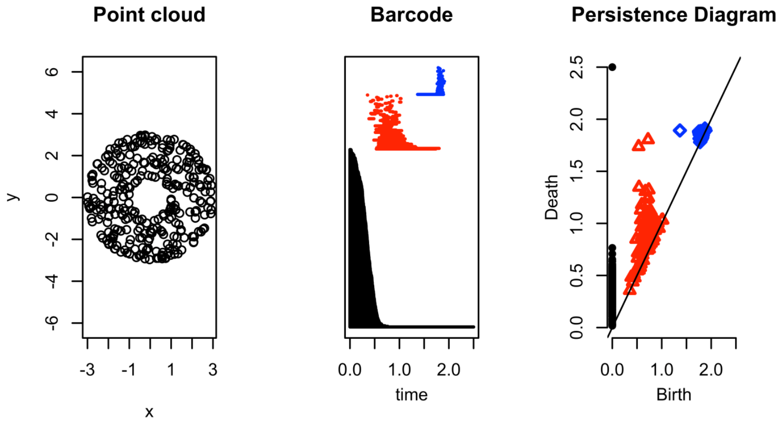

FIGURE 4.5: Barcode and persistence diagram of a point cloud of a torus.

persistence diagram are able to identify one connected component and two small loops despite the presence of noise.

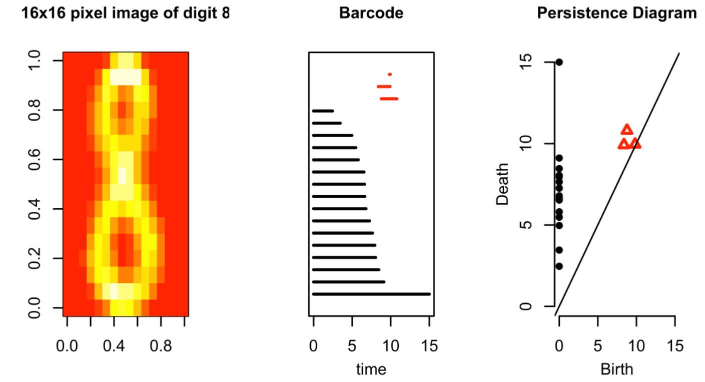

FIGURE 4.6: Barcode and persistence diagram of a point cloud of hand written digit.

We have shown some examples illustrating the application of the theory of persistent homology on point cloud data via barcodes and persistence diagrams and how persistence is useful in the sense that it is robust when measuring topological information of point clouds despite perturbations.

Now we move on to some examples illustrating the use the Mapper algorithm and produce some topological network models of point clouds.

4.2 Mapper on shapes of known topology

We apply Mapper on a point cloud of a noisy circle using the first SVD as a filter function and the Euclidian metric. The clustering method is single linkage, the cover type is balanced with 18 intervals and 40% overlap. We notice that Mapper is able to recover the loop of the circle despite the added noise, see Figure 4.7.

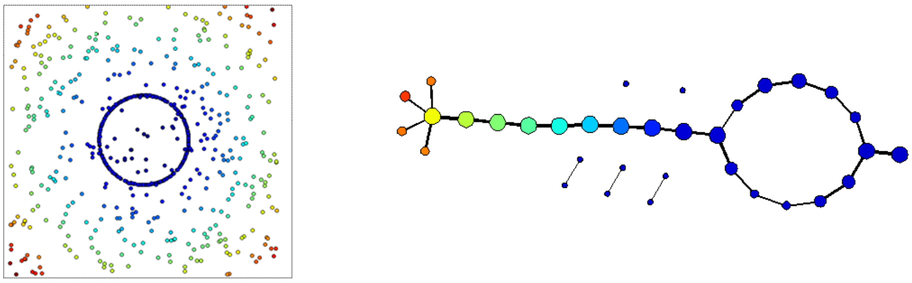

FIGURE 4.7: Topological summary of a sample of a noisy circle colored by the first SVD filter function.

We apply Mapper on a point cloud sampled from a torus in $\mathbb{R}^3$ using the first and second SVD as filter functions and the Euclidian metric. The clustering method is single linkage, the cover is balanced with 18 intervals. Then we color the topological network using the first and the second SVD, see Figure 4.8.

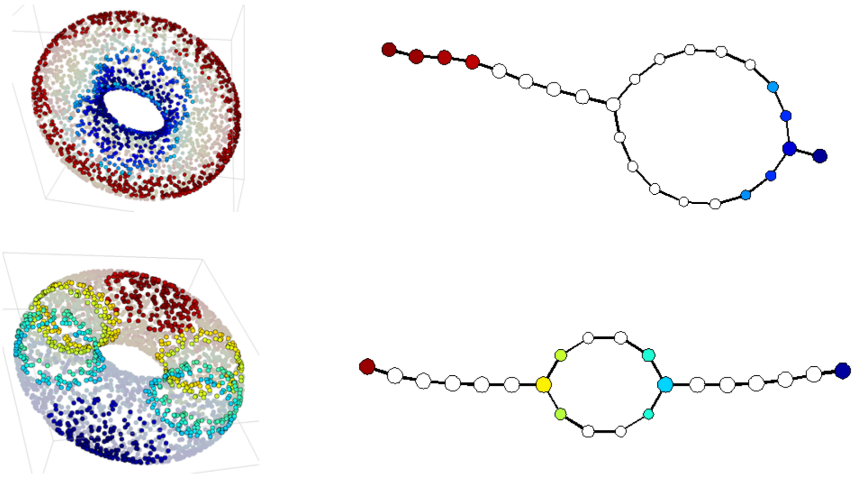

FIGURE 4.8: Topological summary of a sample of torus colored by first SVD filter function (top) and second SVD (bottom).

Another example is to apply both topological mapping and persistent homology on a point cloud known as the chain link or the intertwined rings dataset.

In Figure 4.9, we show that Mapper is able to recognize two connected components and two loops and in Figure 4.10 the persistence diagram yields the same result.

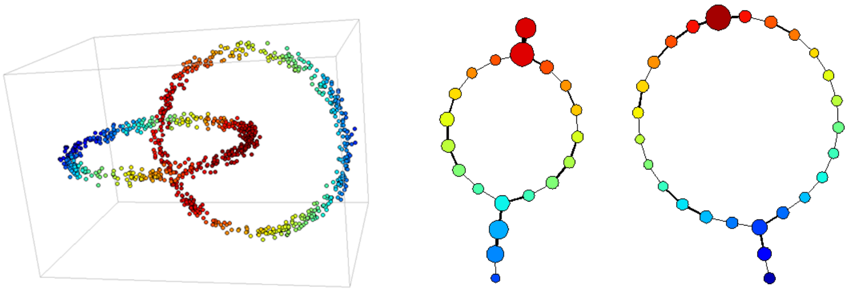

FIGURE 4.9: Topological summary of a sample of the chain link data colored by the SVD filter function.

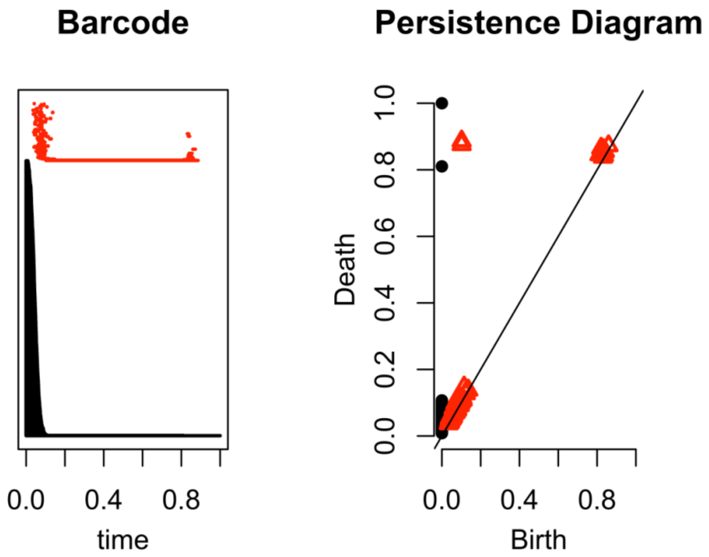

FIGURE 4.10: Barcode and persistence diagram of point cloud of the chain link dataset.

4.3 Mapper on a 3D shape

Here we apply Mapper on a freely available 3D graphics test model known as the Stanford bunny developed at Stanford University (Laboratory, 2015). We randomly sample 5000 point from the original dataset then we apply Mapper using the first SVD filter function together with the Chebyshev metric. The clustering method is single linkage, the cover type is uniform 1-d with 11 intervals and 50% overlap.

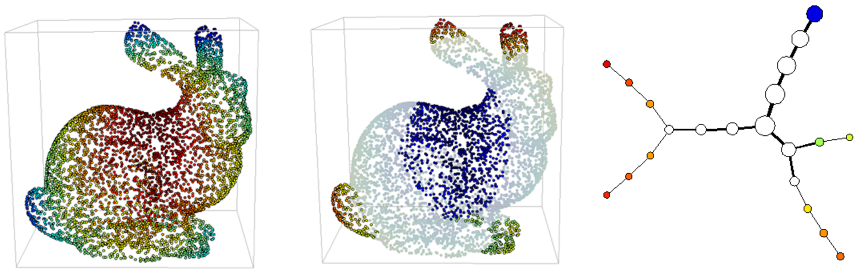

FIGURE 4.11: Topological summary of a sample of 5000 randomly chosen points of the original Stanford bunny point cloud coloured by the first SVD filter function.

Now we randomly select a much less sample size than the previous one, namely 500 points. Similarly we apply Mapper with the same parameter choices as before. Notice that the topological network model is almost identical despite reducing the sample size from 5000 to 500 data points.

Despite the decrease in sample size, Mapper still is able to recover the skeleton of the shape accurately.

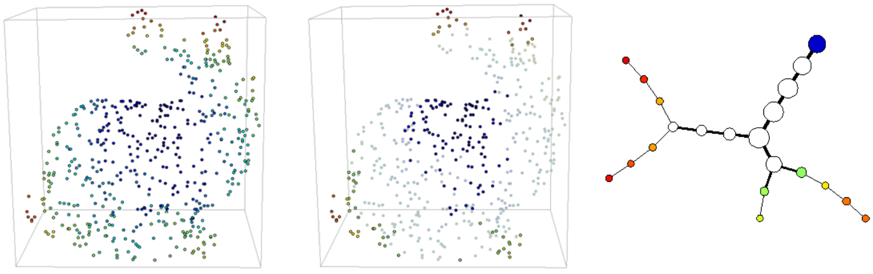

FIGURE 4.12: Topological summary of a sample of 500 randomly chosen points of the original Stanford bunny point cloud coloured by the first SVD filter function.

4.4 Application on a marketing research dataset

The dataset in this application comes from a marketing study of customer purchases of orange juice at five supermarkets (Foster, 1998). The dataset consists of 1070 observations on 17 predictors, see Table 4.1. The dependent variable has two values which are the two types of orange juice that a customer has purchased either, Citrus Hill (CH) or Minute Maid (MM).

TABLE 4.1: A marketing research dataset of customer purchases of two brands of orange juice at five stores.

Purchase	A factor with levels CH and MM
WeekofPurchase	Week of purchase
StoreID	Store ID
PriceCH	Price charged for CH
PriceMM	Price charged for MM
DiscCH	Discount offered for CH
DiscMM	Discount offered for MM
SpecialCH	Indicator of special on CH
SpecialMM	Indicator of special on MM
LoyalCH	Customer brand loyalty for CH
LoyalMM	Customer brand loyalty for MM
SalePriceMM	Sale price for MM
SalePriceCH	Sale price for CH
PriceDiff	Sale price of MM less sale price of CH
PctDiscMM	Percentage discount for MM
PctDiscCH	Percentage discount for CH
ListPriceDiff	List price of MM less list price of CH

Often the first step before doing statistical analysis is to look at the data at hand and be familiar with it through some exploratory graphical analysis before starting fitting models. For example, we learn from the plots in Figure 4.13 that the majority of purchases are Citrus Hill (CH) and that store 5 has the largest share of customer purchases. Also there are differences in the purchase frequency across the five supermarkets.

Furthermore, we might be interested in counting the number of customer purchases of CH and MM across the 52 weeks of purchase in order to see how sales develop with time or how the number of discount coupons offered develops with time, see Figure 4.14.

The prices for CH across the five supermarkets are lower than those for MM. In Figure 4.15 we notice how sale prices for CH across the five stores are lower than those for MM. Also Store 5 has more special offers for CH compared to other stores; on the other hand store 5 does not offer that many specials for MM.

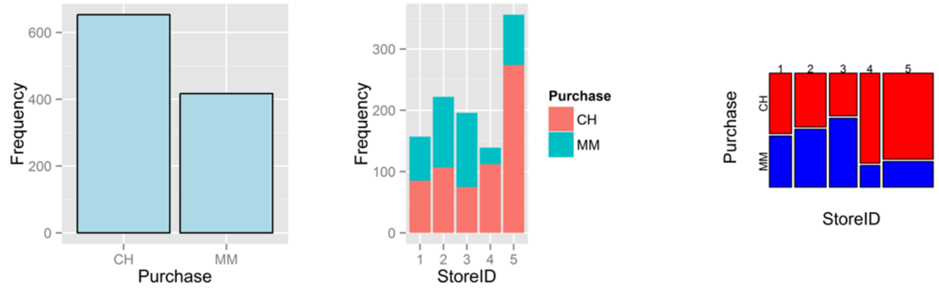

FIGURE 4.13: Some exploratory graphical analysis using bar plots and a mosaic plot.

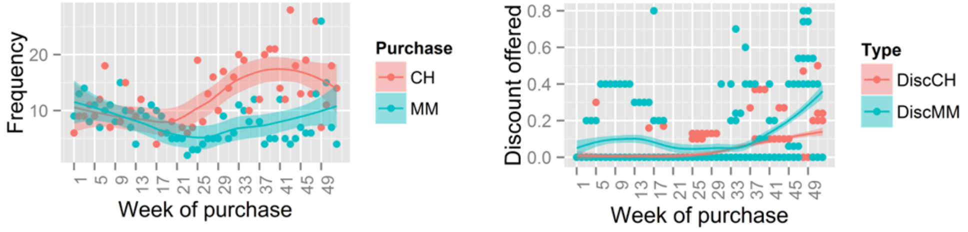

FIGURE 4.14: A plot of the number of customer purchases (left) and the discount offers across the 52 weeks (right).

Among the predictors of the dataset there are two predictors LoyalCH and LoyalMM that are perfectly correlated since both predictors add to one. Consequently as the loyalty of a customer to CH increases, the loyalty to MM decreases. There are other predictors that are almost perfectly correlated such as the percentage discount for CH, namely PctDiscCH and the discount offered for CH, i.e.DiscCH, see Figure 4.16. Similarly PctDiscMM and DiscMM are almost perfectly correlated. Before applying statistics directly to data, one should avoid multicollinearity by removing the predictors LoyalMM, PctDiscCH and PctDiscMM.

We are now going to approach the problem in two ways; first applying statistics directly on the dataset and second applying Mapper as a pre-processing step before applying statistics on the shape of the dataset.

4.4.1 First approach using only statistics

At this point we are going to implement a statistical learning method known as the penalized logistic regression which is a widely used classification method when the response is categorical, for example with two possible outcomes $\{1, 0\}$. The two possible outcomes in this example are CH and MM.

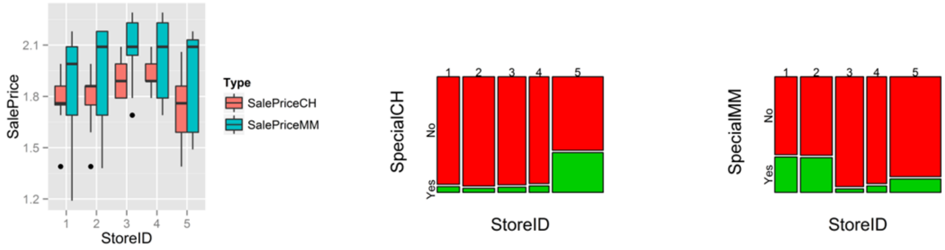

FIGURE 4.15: Some exploratory graphical analysis of using box plot and mosaic plots.

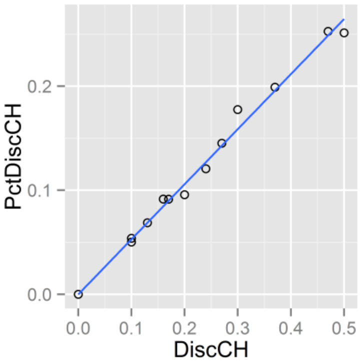

FIGURE 4.16: Two almost perfectly correlated predictors.

We use the L1 penalty which tends to shrink many regression coefficients exactly to zero which in return makes it possible to do feature selection in order to find the predictors that are most informative. The shrinkage also prevents overfitting due to possible collinearity of the predictors or in the case of high dimensional datasets.

The whole path of variables with coefficient shrinkage is shown in the left plot of Figure 4.17. We then apply 10-fold cross validation in order to find the optimal value of the tuning parameter λ that gives minimum mean cross-validation error, see Figure 4.17 (below).

The misclassification error is a measure of the fraction of instances misclassified by the model. The fraction deviance explained can be thought of as roughly speaking the amount of variability in the response explained by the predictors; similar idea to the measure of goodness of fit R^2 explained in section 3.2 (regression models). For more details two comprehensive references are (Hastie, 2009) and (Hastie, 2015).

We select the first λ in order to obtain the most regularized model such that the error is within one standard deviation from the minimum λ. Hence it is a slightly more restricted model that does almost

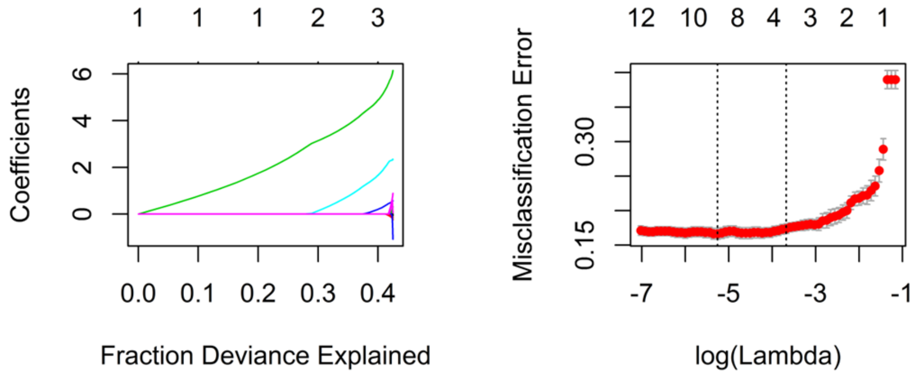

FIGURE 4.17: The fitted object using an L1 absolute value lasso penalty $\alpha = 1$ (left) and The cross validation curve with error bars and the optimal value of λ in dashed line (right).

as well as the minimum. The resulting predictors and their corresponding coefficients are shown in Table 4.2. The result suggests a model with LoyalCH, PriceDiff and an indicator variable for StoreID5. Note that the positive coefficients imply an increasing preference for purchasing CH.

We now try to interpret the resulting coefficients in Table 4.2. The increase in brand loyalty for CH, i.e. the covariate LoyalCH implies an increasing preference for purchasing CH. Also since the price difference covariate PriceDiff is defined as sale price of MM minus sale price of CH, the model therefore suggests that CH purchases are likely to happen when the sale price of MM is higher than the sale price of CH, in other words customers are likely to go for the lower sale price. Store number five stands out from the other supermarkets. This is plausible since it has most special offers and best sale prices for CH, see the mosaic plots in Figure 4.15.

TABLE 4.2: Result obtained using the first λ within one standard deviation from the minimum λ.

Predictor	Coefficient
LoyalCH	5.135
PriceDiff	1.874
StoreID5	0.296

It is quite plausible to consider interactions between variables since predictors such as discount offered or special offers seems to be masked so far. Sometimes the influence of one predictor is linked to the influence of another predictor; for now we reconsider the logistic model and reformulate it in terms of the previously used predictors and their pairwise interactions. The dimension of $\mathbf{X}$ will then become $\binom{p}{2}$ plus the number of the original predictors p.

The whole path of variables with coefficient shrinkage and the cross validation curve are shown in Figure 4.18. We use the first λ in order to obtain the most regularized model such that the error is within one standard deviation from the minimum λ.

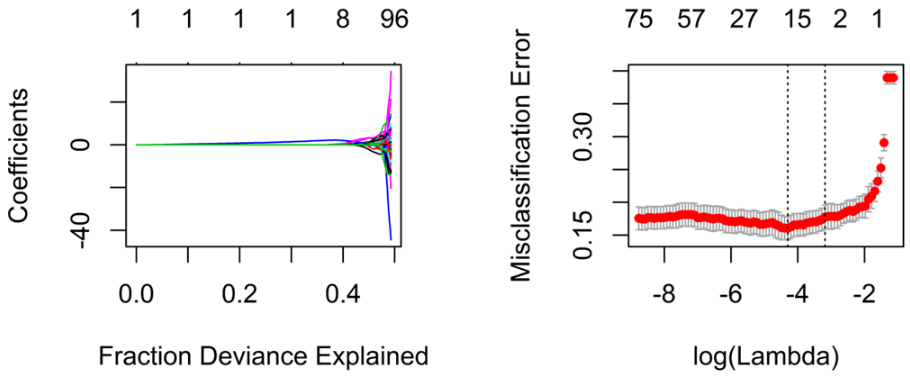

FIGURE 4.18: The fitted object using an L1 absolute value lasso penalty $\alpha = 1$ (left) and The cross validation curve with error bars and the optimal value of λ in dashed line (right).

The resulting predictors and their corresponding coefficients are shown in Table 4.3. The result suggests a model with LoyalCH and some interactions for instance StoreID5*ListPriceDiff suggests that the effect of lower prices for CH in store 5 relative to other supermarkets are pushing the sales of CH. We notice that the effect of some predictors that did not show up in the previous logistic model is now revealed, for example the effect of discount coupons and special offers DiscCH*SpecialCH. A possible interpretation is that customers are less impressed by discount coupons and special offers if the list prices were expensive. Interpreting interactions can be difficult sometime but in general an interaction between two predictors means that the effect of one of them cannot be separated from the other.

We are now going to show the second approach of first using Mapper as a pre-processing step before using the penalized logistic regression on the shape of the data.

TABLE 4.3: Result obtained using the first λ within one standard deviation from the minimum λ.

Predictor	Coefficient
LoyalCH*SalePriceMM	2.063
StoreID5*ListPriceDiff	0.718
LoyalCH	0.563
LoyalCH*ListPriceDiffalCH	0.487
DiscCH*SpecialCH	0.123
WeekofPurchase*PriceDiff	0.019
WeekofPurchase*SpecialMM	-0.002

4.4.2 Second approach using Mapper and penalized logistic regression

We now apply Mapper on the dataset and we include all the predictors. We use the first metric SVD as filter function and the Chebyshev metric. The clustering method is single linkage, the cover is balanced with 10 intervals and 40% overlap and the cut-off is first gap with relative width 0.4.

We first colour the topological summary of the point cloud using the filter function and then we colour by CH; finally we colour by MM, see Figure 4.19.

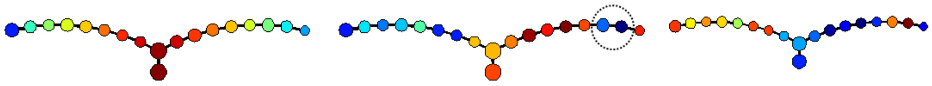

FIGURE 4.19: Topological summary of the dataset coloured by the first SVD filter function (left), coloured by PurchaseCH (center) and by PurchaseMM (right).

We examine the flares shown in the network model and just by inspecting the network visually, we notice a gap of blue nodes surrounded by red nodes; see the dotted black circle in Figure 4.19. This gap is filled when the network is colored by PurchaseMM as we see in Figure 4.19 but the fact that there are two blue regions indicates that when a customer decides not to purchase CH is not always for the same reason; instead there are possible different reasons which suggests to try to determine what possible predictors associated with MM that could fill in that gap. Now some possible candidates could be for example brand loyalty for MM LoyalMM, the sale price SalePriceMM, the discount coupons DiscMM or the special offers SpecialMM.

We notice in Figure 4.20 that both DiscMM and/or SpecialMM are possible candidates; so we have a rough idea of what predictors

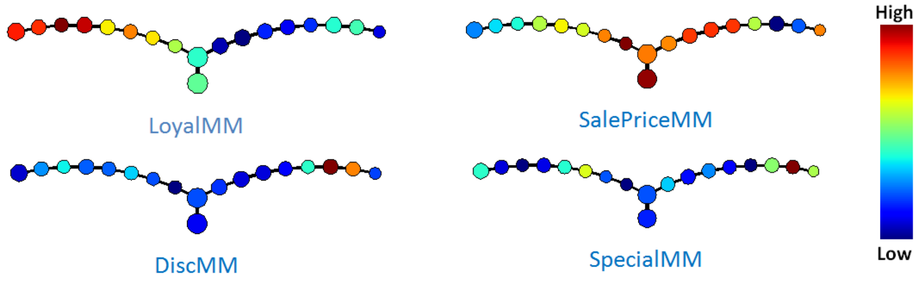

FIGURE 4.20: Topological summary of the dataset colored by four different predictors written under each graph.

to expect to show up having an association with the response variable.

We proceed by implementing the penalized logistic regression in two steps.

- *Step one* is to compare the blue nodes on the right flare with the left flare.
- *Step two* is that we compare the blue nodes with the surrounding red nodes on the same flare, see Figure 4.21.

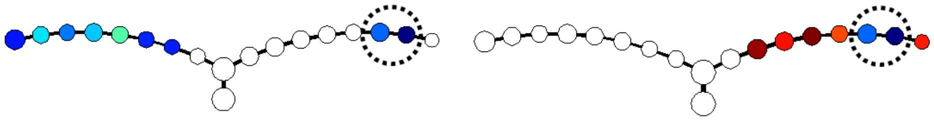

FIGURE 4.21: The group of the two blue nodes is indicated by a black dotted circle and is compared to first the other left hand flare and then to its surrounding red nodes of the right hand flare.

Step one: The whole path of variables with coefficient shrinkage and the cross validation curve are shown in Figure 4.22. We use the first λ in order to obtain the most regularized model such that the error is within one standard deviation from the minimum λ.

We use logistic regression and cross validation to distinguish between the two blue nodes indicated by the dotted black circle and the other flare.

The resulting predictors and corresponding coefficients are shown in Table 4.4. We notice that the result is almost identical to Table 4.2, namely when we applied logistic regression to the whole dataset. The response in the two cases is different, namely in the first case the

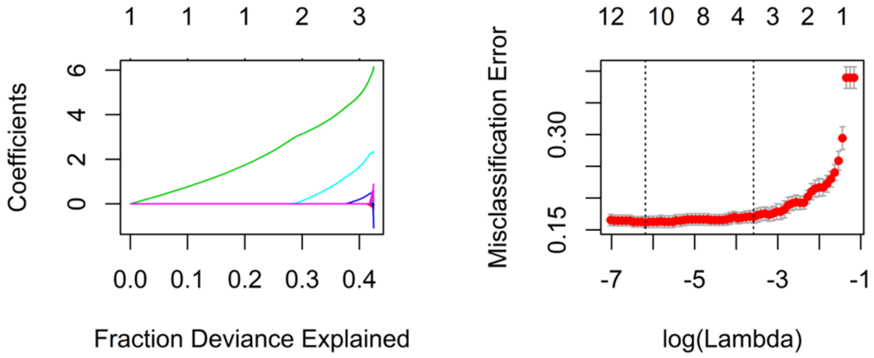

FIGURE 4.22: The fitted object using an L1 absolute value lasso penalty $\alpha = 1$ (left) and The cross validation curve with error bars and the optimal value of λ in dashed line (right).

TABLE 4.4: Result obtained using the first λ within one standard deviation from the minimum λ.

Predictor	Coefficient
LoyalCH	5.047
PriceDiff	1.801
StoreID5	0.264

response was Purchase of two levels CH and MM and in the second case the response is groups of cluster or nodes.

Usually at this stage one begins to interpret the resulting predictors using the predictors and their corresponding coefficients. However before doing that we could color the network by the resulting predictors in order to do better interpretation.

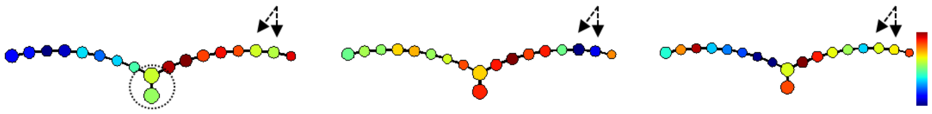

FIGURE 4.23: The network of the dataset coloured by LoyalCH (left), coloured by PriceDiff (centre) and by StoreID5 (right).

In view of Table 4.4 and Figure 4.23, we can interpret the two nodes indicated by the two black arrows as a group of purchases done by customers characterized by being moderately loyal for the brand CH. Those customers prefer to buy the competitive brand MM when the price difference PriceDiff (sale price MM $-$ sale price CH) is quite low.

We also notice a group of purchases (in the dotted black circle in Figure 4.23) done by customers that are also characterized by being moderately loyal for CH but here the price different is not so low; those customers are likely to prefer the lower sale price, namely CH. Since the dataset represent observed purchases, it is possible that one and the same customer is in both green nodes indicated by the black arrows or the dotted black circle. That particular customer is moderately loyal for CH but depending on the PriceDiff situation, he or she chooses between buying CH or MM. Furthermore Store 5 is distinguished as it has mostly special offers compared to the other stores and it also has the best sale prices for CH. On the other hand it offers some specials for MM and made sales for MM during 52 weeks of purchase.

Step two: we now want to distinguish between the two blue nodes indicated by the dotted black circle and the surrounding red nodes on the same flare. The whole path of variables with coefficient shrinkage and the cross validation curve are shown in Figure 4.24. We use the minimum λ in order to obtain the most regularized model.

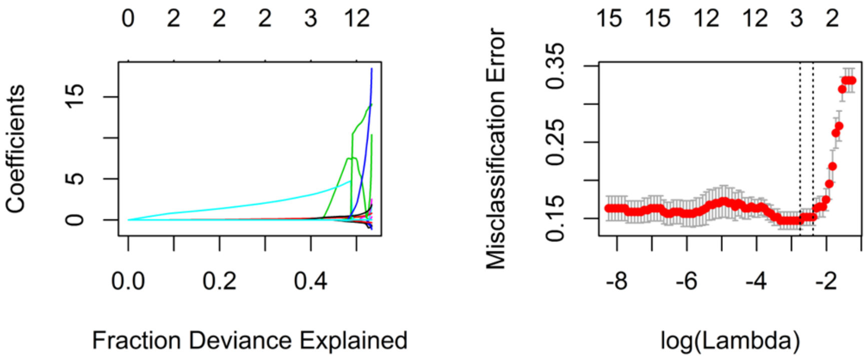

FIGURE 4.24: The fitted object using an L1 absolute value lasso penalty $\alpha = 1$ (left) and The cross validation curve with error bars and the optimal value of λ in dashed line (right).

TABLE 4.5: Result obtained using the first λ within one standard deviation from the minimum λ.

Predictor	Coefficient
DiscMM	3.065
WeekofPurchase	0.191
SpecialMM	0.126

In view of Table 4.5 and Figure 4.25, we add to the previous interpretation the following: purchases done by customers characterised

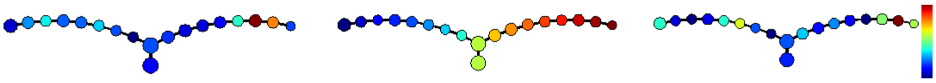

FIGURE 4.25: The network of the dataset coloured by DiscMM (left), coloured by WeekofPurchase (centre) and by SpecialMM (right).

by being moderately loyal for CH are likely to prefer buying the competitive brand MM when there are discount coupons for MM. Discount coupons for MM are offered more frequent relative to CH towards the end of the 52 weeks of purchase; that is likely the reason why WeekofPurchase is significant, see the time series plot in Figure 4.14. Also special offers for MM play a role but not as much as discount coupons. This is possibly because MM specials are offered mostly at Store 1 and 2, see mosaic plots in Figure 4.15.

4.4.3 Persistent homology on the dataset

We experiment with persistent homology by applying it onto the dataset. Figure 4.26 shows the barcode and the persistence diagram of the dataset with all predictors included.

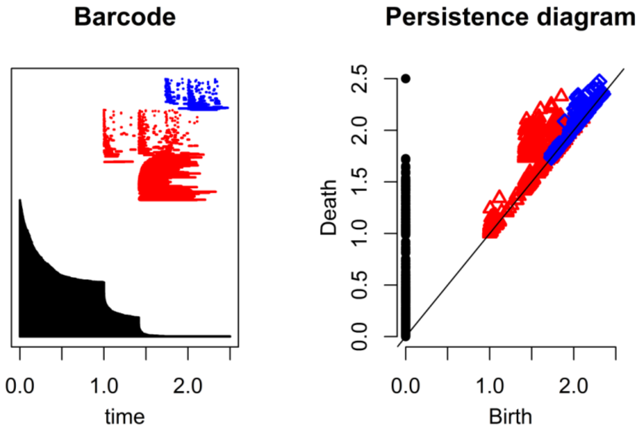

FIGURE 4.26: Barcode and persistence diagram of all the predictors in the dataset.

Next we remove some predictors that are considered to be noise in order to see whether the barcode or the persistence diagram are going to change.

We remove PriceCH, PriceMM, SalePriceCH, SalePriceMM, PctDiscCH, PctDiscMM and ListPriceDiff. We notice that both barcode and persistence diagram did not change much, see Figure 4.27. Finally we try to remove some of the important predictors for example PurchaseCH, PurchaseMM, StoreID5 and PriceDiff from the dataset. We notice how the structure changes significantly in Figure 4.28.

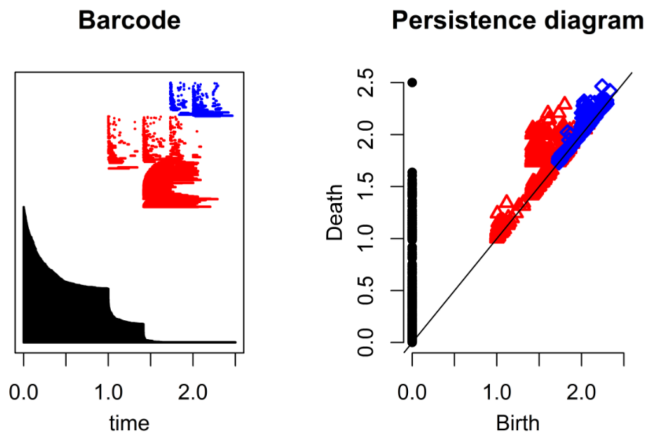

FIGURE 4.27: Removing predictors PriceCH, PriceMM, SalePriceCH, SalePriceMM, PctDiscCH, PctDiscMM and ListPriceDiff considered being noise.

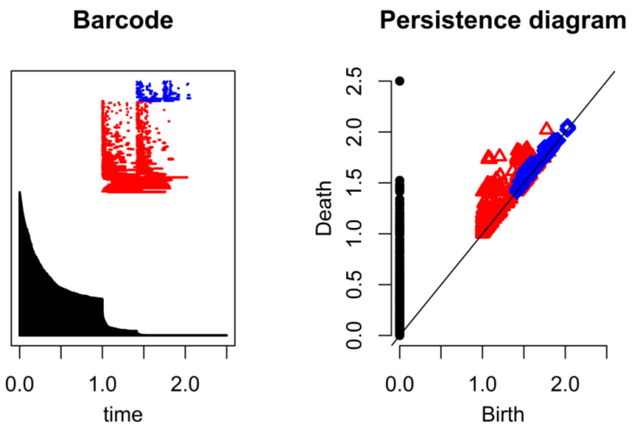

FIGURE 4.28: Removing predictors that are considered Significant.

4.4.4 Summary of the previous approaches

We approached the marketing research dataset using only a classification technique to measure the relationships between the dependent variable and the set of predictors. Then we approached the problem again using Mapper as a pre-processing organizing step before implementing classification. That led us to the same result as the penalized logistic regression despite using only a portion of the dataset suggested by the topological network.

Then we considered the pairwise interactions between the set of predictors and we tried to interpret the interactions. Afterwards we applied Mapper before implementing classification on the flares of the topological network and in return we obtained a clear result that we could easily interpret without considering pairwise interactions.

4.5 Application on workers wage dataset

The dataset in this application is about a group of males from the Atlantic region of the USA. The response variable is workers wage and is a continuous variable. The dataset consist of 3000 observations on 12 predictors. The response variable and the predictors are shown in Table 4.6.

TABLE 4.6: A data set from the Atlantic region of the USA of a group of male workers.

year	Year that wage information was recorded
age	Age of worker
sex	Gender
maritl	A factor with levels indicating marital status: 1) Never Married 2) Married 3) Widowed 4) Divorced 5) Separated
race	A factor with levels indicating race: 1) White 2) Black 3) Asian 4) Other
education	A factor indicating education level : 1) < HS Grad 2) HS Grad 3) Some College 4) College Grad 5) Advanced Degree indicating education level
region	Region of the country (mid-atlantic only)
jobclass	A factor with levels indicating type of job: 1) Industrial 2) Information
health	A factor with levels indicating health level of a worker: 1) <=Good 2) >=Very Good
health-ins	CA factor with levels indicating whether worker has health insurance 1) Yes 2) No
logwage	Log of workers wage
wage	Workers raw wage

Some predictors in Table 4.6 can be omitted such sex or region since the study is about a group of only males from the Atlantic region.

We implement the second approach directly, namely we apply Mapper on the dataset together with the penalized logistic regression and if necessary we do some exploratory graphical analysis.

We apply the secondary metric SVD as filter function and the Chebyshev metric. The clustering method is complete linkage, the cover is balanced with 16 intervals and 40% overlap. We color the network using first the filter function and then we color it by workers wage, see Figure 4.29.

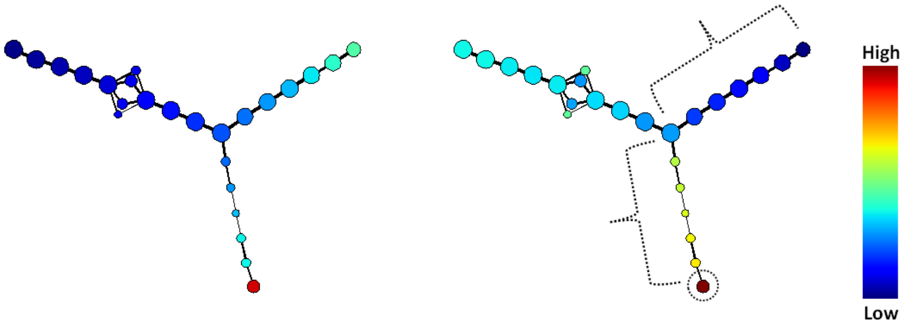

FIGURE 4.29: Topological summary of the dataset colored by the secondary SVD filter function (left) and colored by wage (right).

First we notice the red cluster in the bottom flare indicated by the dotted black circle. This suggests that there is a cluster of high wages quite different from the rest population. We can furthermore color the topological summary with two more interesting predictors, namely year and age, see Figure 4.30.

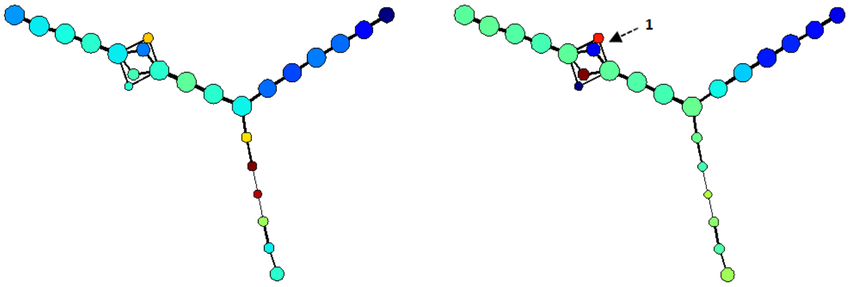

FIGURE 4.30: Topological summary of the dataset colored by year (left) and colored by age (right).

Before applying logistic regression, we might be interested in examining the previous networks to be familiar with the data and to generate some qualitative hypothesis, for instance there seems to be a low share of high-wage works while the majority of wages are low or medium.

We make a plot of wage versus age (shown in Figure 4.31) and we highlight the group of clusters indicted by number 1 from Figure 4.30. We notice that the density of points is decreasing as we move towards higher wages.We can also make a plot of wage versus year as in Figure 4.31. We notice that there is a slow steady increase of wages across the years and also that wages increase with age but until about 60 years old after which a decrease occur. The wage versus age plot seems to be quite dense in the middle and as we move towards its boundaries the point cloud becomes less dense; in particular as we move up towards higher wages.

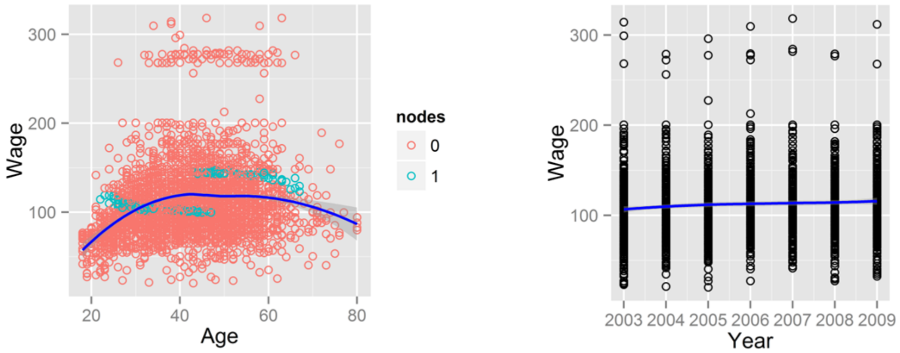

FIGURE 4.31: Scatter plot of wage vs age, nodes 1 in Figure 4.30 are highlighted (left) and a scatter plot of wage vs year (right).

4.5.1 The approach using Mapper and logistic regression

Now we can apply the penalized logistic regression on the two flares of the network of Figure 4.29.

The whole path of variables with coefficient shrinkage and the cross validation curve are in Figure 4.32. We use the minimum λ in order to obtain the most regularized model; the result is shown in Table 4.7.

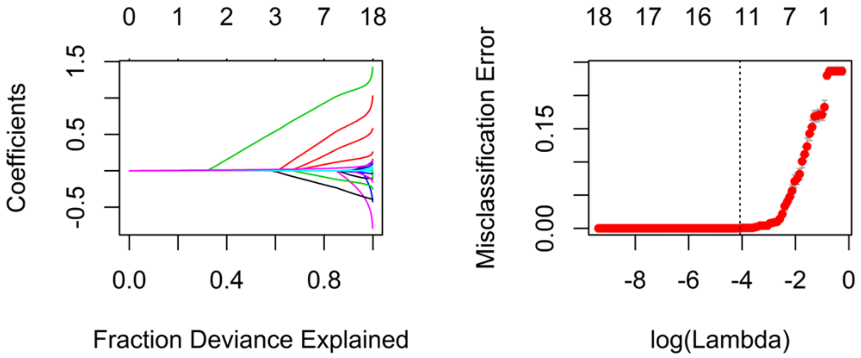

FIGURE 4.32: Scatter plot of wage vs age, nodes 1 in figure 2 are highlighted (left) and a scatter plot of wage vs year (right).

The resulting predictors and the corresponding coefficients in Table 4.7 suggest that workers who earn high wages have either an

TABLE 4.7: Result obtained using the minimum λ.

Predictor	Coefficient
education5. Advanced Degree	1.100
education4. College Grad	0.647
maritl2. Married	0.380
health-ins1. Yes	0.151
wage	0.047
jobclass2. Information	0.015
year	0.004
health2. >=Very Good	0.003
age	0.001
health1. <=Good	-0.003
jobclass1. Industrial	-0.015
maritl1. Never Married	-0.062
health-ins2. No	-0.151
education1. < HS Grad	-0.168
education2. HS Grad	-0.305

advanced degree or a college degree and they are likely to be married and have health insurance; their job class is in the information sector and their health status is classified as more than or equal to very good. Finally their wages are likely to increase with time and age.

On the other hand workers who are associated with relatively low wages are likely to have a high school degree or less; they are likely not to be married and nor health insurance; their job class belong to the industrial sector and their health status is classified as less than or equal to good. Some exploratory bars plots as in Figure 4.33 show how wages on average increase with the level of education. We can also plot wages across education levels for each race group. At the education level of less than high school, the Asian group seems to have the lowest wage relative to the other groups. On the other hand we notice that at the advance degree education level, both the white and the Asian groups seem to have the highest wages relative to the rest.

Now we might be interested in analyzing wages for a specific group, for example we might be interested in coloring the network of the dataset by race2.black as in Figure 4.34. We notice that workers in that group are associated with low and medium wages, notice the red hot spots indicated by number 1 and 2.

Suppose we are interested in analyzing the partition race2.black, for example in order to discover some distinguishing factors between those who earned more and those who earned relatively less. We apply logistic regression to distinguish between the nodes indicated by

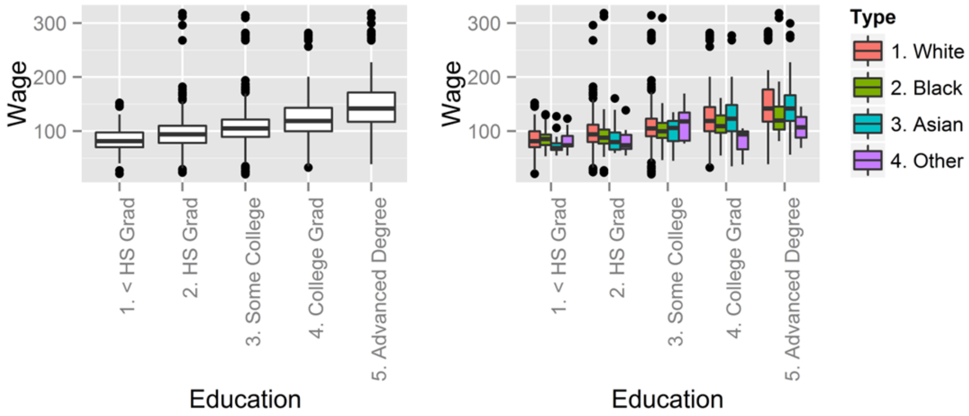

FIGURE 4.33: Some exploratory graphical analysis of using box plots.

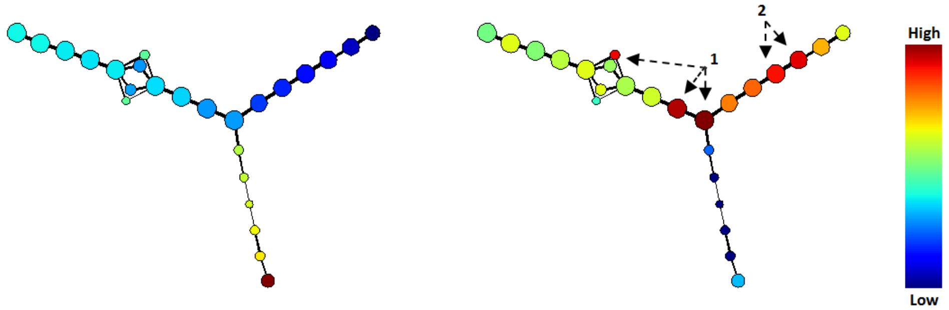

FIGURE 4.34: Topological summary of the dataset colored by wage (left) and colored by race2.black (right).

numbers 1 and 2 in Figure 4.34. The whole path of variables with coefficient shrinkage and the cross validation curve are shown in Figure 4.35. We use the minimum λ in order to obtain the most regularized model; results are shown in Table 4.8.

TABLE 4.8: Result obtained using the minimum λ.

Predictor	Coefficient
wage	0.142
health-ins1. Yes	0.102
education3. Some College	0.099
age	0.035
health-ins2. No	-0.101
education1. < HS Grad	-0.259

The resulting predictors and the corresponding coefficients in Table 4.8 suggest that workers from the black community of the Atlantic region of the USA who earn relatively high wages are likely to

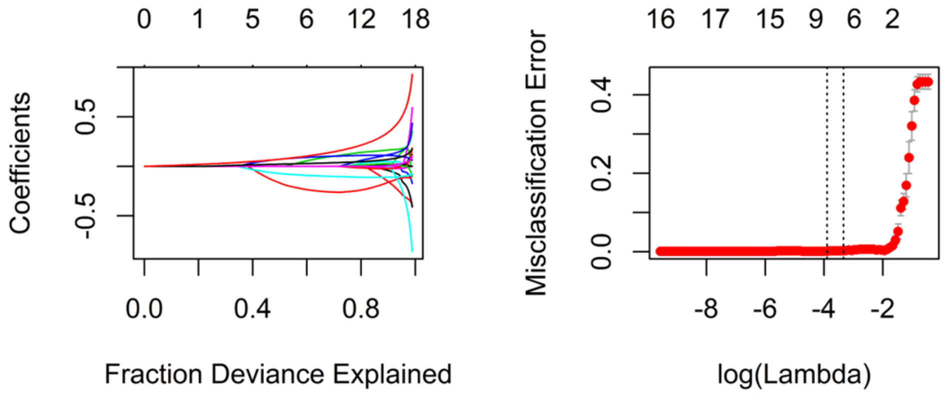

FIGURE 4.35: The fitted object using elastic net $\alpha = 0.5$ (left) and The cross validation curve with error bars and the optimal value of λ in dashed line (right).

have health insurance, some college degree and they are likely to be older. By contrast those who earned relatively low wages are likely to be those who did not graduate from high school or did not have health insurance.

4.5.2 Persistent homology on the dataset

We plot in Figure 4.36 the barcode and the persistence diagram of the dataset with all the predictors included. Then we remove some predictors such as the indicator variable maritl with levels never married, married, widowed, divorced and separated.

We notice in Figure 4.37 that both barcode and persistence diagram did not change that much; however when we try to remove one important predictor such as age we notice a significant change in the structure of the barcode and the persistence diagram, see Figure 4.38.

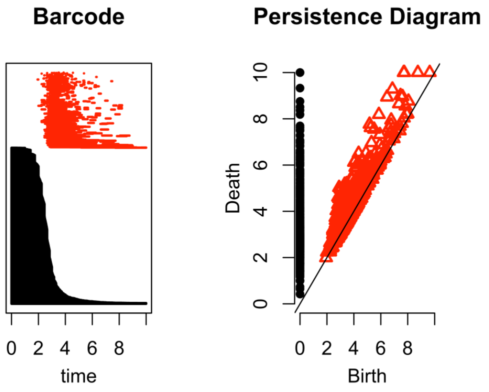

FIGURE 4.36: Barcode and persistence diagram of all the predictors in the dataset.

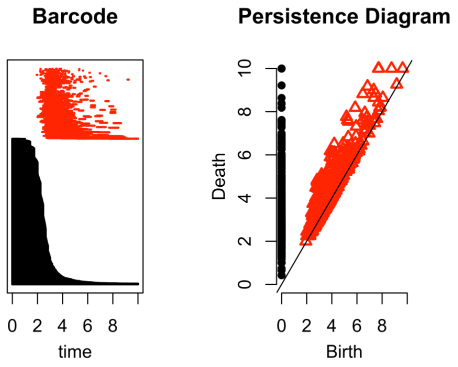

FIGURE 4.37: Barcode and persistence diagram of all the predictors of the dataset except maritl.

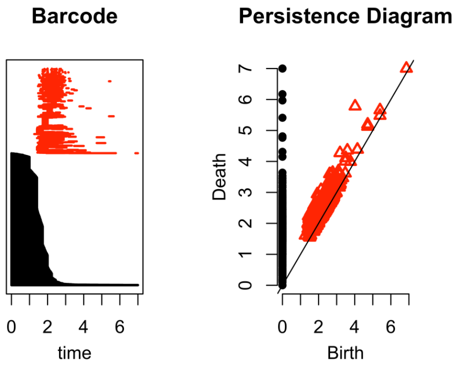

FIGURE 4.38: Barcode and persistence diagram of all the predictors of the dataset except age.

5. Conclusions and Discussion

5.1 Summary

We began section 4 (Experimenting with actual data) with applications of persistent homology and the Mapper algorithm on several point clouds. We have shown that homology can measure topological features such as the number of connected components, loops or voids and we illustrated computing barcodes and persistence diagrams of the Rips filtration constructed on several point clouds. We have also shown how Mapper is able to recognize random samples of shapes of known topology despite the presence of noise.

Mapper is able to represent a large number of data points in high dimensional space by a two dimensional network model that is easy to visualize. The dataset of the intertwined rings demonstrated how both Mapper and persistent homology were able to recover the same topological features, namely two connected components and two loops. Then, regarding the Stanford bunny dataset, we started with a sample size of 5000 uniformly distributed data points and produced the corresponding topological network model. Afterwards we reduced the sample size dramatically to 500 data points and still obtained a very similar topological summary. Mapper was able to recover the skeleton of the shape accurately despite the large decrease in sample size.

We then started to analyze a real world application of a marketing research dataset. First, we approached the problem using only logistic regression as a classification technique to measure the relationships between a categorical dependent variable and a set of predictors. We, additionally, had to consider pairwise interactions between the set of predictors since there might be situations where the effect of one predictor cannot be separated from the other. We have seen that interpreting interactions can be difficult and often confusing.

In contrast, we applied statistics not directly on the data but instead on the shape of the data; namely, in the second approach we used Mapper as a pre-processing organizing step before implementing classification on the flares of the topological summary and in return we obtained a clear result that we could easily interpret without the need to consider pairwise interactions between the predictors.

The second real world application considered workers wage. The response was a continuous dependent variable; however we still were able to use the same approach, namely using Mapper then applying logistic regression on the flares despite the difference in the response type in the two datasets. In return we obtained a set of associations that we straightforwardly were able to interpret and by using cross-validation as a model evaluation method, we were able to generalize those associations to hold on previously unseen data.

Furthermore applying Mapper on the workers wage dataset as a pre-processing step helped us in generating hypotheses and in performing local analysis on a particular group of interest which directed us to extract new insights that we were not looking for initially.

5.2 Discussion

The first issue to discuss is adjusting the Mapper parameters such as the choice of a filter function, type of covering or a clustering algorithm; for example, regarding the dataset of the intertwined rings shown in section 4 (Experimenting with actual data) we produced a barcode, a persistence diagram and a Mapper topological summary which all recognized two connected components and two loops. It is reasonable to think that persistent homology is able to help us in restricting our parameter choices in the Mapper framework since persistence ignores features that appear only for a short period of time and therefore it is robust to noise and outliers.

Another issue to discuss is standardization; consider a situation where we have one variable X_1 defined as the population size measured in number of inhabitants of a city and another variable X_2 defined to be the number of schools in that city. Since X_1 is obviously many orders of magnitude greater than X_2, the variable X_1 will dominate the result and X_2 will be almost ignored. Therefore it is often reasonable to standardize a set of variables by for example equalizing the ranges of the variables or by equalizing their variances using the formula $\tilde{X_k} = X_k/Var\left(X_k\right)$. However implementing standardization roughly means that all predictors are then equally important but there can be some situations where we would like to preserve the information contained in the relative scaling.

Finally, a personal view about data analytics in industry is as follow; in general, all companies tend to be interested in finding better ways of making informed business decisions based on their data. It seems to me that there are two opposed extremes in this issue. One extreme is to rely heavily on sophisticated machine learning algorithms and the other extreme is to rely on experts with a vast amount of domain knowledge. A mixture between the two extremes might be a good solution in the sense that using a topological mapping tool such as Mapper to convert a large amount of data consisting of numbers and text into a two dimensional graph that is easy to visualize can reinforce human cognition and creativity. The viewers of a topological network model, for example, a person with domain knowledge or another with machine learning knowledge, could work together to generate hypotheses or to discover new insights that could assist in solving challenging problems.

Bibliography

A. Babu, D. M. a. (2013). "Python Mapper: An open-source toolchain for data exploration, analysis and visualization". In: *The web site http://danifold.net/mapper.*

Balakrishnan, N. (2014). "Methods and Applications of Statistics in Clinical Trials". In: *Hoboken : Wiley.*

Borovkov, A. A. (2013). "Probability Theory". In: *Springer London.*

Carlsson, G (2009). "Topology and Data". In: *Bull. Amer. Math. Soc. 46 (2009), 255-308.*

Carlsson, G. and F. Mémoli (2010). "Characterization, stability and convergence of hierarchical clustering methods". In: *The Journal of Machine Learning Research, vol. 11, pp. 1425-1470.*

Demmel., J. W. (1997). "Applied Numerical Linear Algebra". In: *Society for Industrial and Applied Math.*

Foster, D. P. (1998). "Business Analysis Using Regression A Casebook:" in: *Springer New York : Imprint: Springer.*

G. Singh, F. Memoli and G. Carlsson (1991). "Mapper: a topological mapping tool for point cloud data". In: *Eurographics symposium on point-based graphics.*

Gut, A. (2012). "Probability A Graduate Course, 2nd ed". In: *Dordrecht : Springer.*

Hastie, T. (2009). "The elements of statistical learning : data mining, inference, and prediction, 2. ed". In: *Springer.*

— (2015). "Statistical Learning with Sparsity The Lasso and Generalizations". In: *Hoboken: Hoboken : CRC Press.*

Husmeier, D. (2006). "Probabilistic Modeling in Bioinformatics and Medical Informatics". In: *Dordrecht : Springer.*

J. K. Brittany T. Fasy Fabrizio Lecci, Clement Maria Vincent Rouvreau. The included GUDHI is authored by Clement Maria Dionysus by Dmitriy Morozov, Michael Kerber PHAT by Ulrich Bauer, and Jan Reininghaus. (2015). "TDA: Statistical Tools for Topological Data Analysis". In: *web site https://cran.r-project.org/web/ packages/TDA/index.html.*

Johnson, R. A. and D. W. Wichern (2013). "Applied Multivariate Statistical Analysis: Pearson New International Edition". In: *Pearson.*

Kroese, D. P. and J. C.C. Chan (2014). "Statistical Modeling and Computation". In: *New York, NY: Springer New York.*

Laboratory, S. U. C. G. (2015). "The Stanford 3D Scanning Repository". In: *web site http://graphics.stanford.edu/data/3Dscanrep/bunny.*

Mac Lane, Saunders (1963). "Homology". In: *Berlin: Springer*.

Müllner, D. (2013). "fastcluster: Fast Hierarchical, Agglomerative Clustering Routines for R and Python," in: *2013, vol. 53, p. 18, 2013-05-29 2013*.

P. Y. Lum G. Singh, A. Lehman T. Ishkanov M. Vejdemo-Johansson M. Alagappan et al. (2013). "Extracting insights from the shape of complex data using topology". In: *Scientific Reports, vol. 3*.

Singh, G. (2008). "Algorithms for topological analysis of data". In: *Stanford University*.

T. K. Dey, F. Memoli and Y. Wang (2015). "Mutiscale Mapper: A Framework for Topological Summarization of Data and Maps". In: *eprint arXiv:1504.03763*.

TRITA -MAT-E 2016:03
ISRN -KTH/MAT/E--16/03--SE

www.kth.se