Using Topological Data Analysis to Process Time-series Data: A Persistent Homology Way

Gang Ma
School of Remote Sensing and Information Engineering, Wuhan University, Wuhan; 430079, China
Corresponding E-mail: angela@cas-harbour.org

Abstract. Topological Data Analysis (TDA) is a novel new and strong-growing method to deal with various data in most areas. And Persistent Homology is one of the most pivotal tools in Topological Data Analysis to acquire topological properties of the data. This article is based on the main mathematics behind Topological and Topological Data Analysis. And it describes how to use the above theories and methods to do the analysis job for time-series data. Moreover, it discusses the further applications of TDA to other domains and the combination of machine learning with Topological Data Analysis. The article outlines the TDA model and principle behind the data set and provides insights into the function of TDA for time-series analysis as well as opportunities for future work.

1. Introduction
Based on the applied algebraic topology, Topological Data Analysis (TDA) is a comparatively new approach to give insights into the complex data sets. It has been successfully applicable to various areas including biology[15], information science, bio-chemistry, brain science[3], business analysis [11] and material science. Among the above applications, it does not provide feature vectors or descriptors of data from their complex geometric configurations directly. The aim of TDA is to decrease the dimensionality of high dimensional data and analyzing the topological properties, structure and shape of the given data and finally clustering complex data. In this case, TDA can help people understanding and predicting links in various kinds of data sets where data are often indicated by point clouds in Euclidean or more general metric spaces. The article introduces the TDA framework and rationale behind the data set and investigates the role of TDA for time-series analysis as well as opportunities for new work.

2. Topological Summaries

2.1 Metric Spaces and Hausdorff Distance
Definition 1(Metric Space). Recall that a metric space $(M, \rho)$ is a set $M$ with a function $\rho : M \times M \to \mathbb{R}_+$ called a distance, such that for any $x, y, z \in M$:
- $\rho(x, y) \geq 0$ and $\rho(x, y) = 0$ if and only if $x = y$
- $\rho(x, y) = \rho(y, x)$
- $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$

Given a metric space $(M, \rho)$, the set $\mathcal{K}(M)$ of its compact subsets can be endowed with the so-called Hausdorff distance.
Definition 2 (Hausdorff Distance). Given two compact subsets \( A, B \subseteq M \), the Hausdorff distance \( d_H(A, B) \) between \( A \) and \( B \) is defined as the smallest non-negative number \( \delta \) such that for any \( a \in A \) there exists \( b \in B \) such that \( \rho(a, b) \leq \delta \) and for any \( b \in B \), there exists \( a \in A \) such that \( \rho(a, b) \leq \delta \). In other words, if for any compact subset \( C \subseteq M \), we denote by \( d(C) : M \rightarrow \mathbb{R} \) the distance function to \( C \) defined by

\[
d(x, C) = \inf_{c \in C} \rho(x, c)
\]

for any \( x \in M \), then one can prove that the Hausdorff distance between \( A \) and \( B \) is defined by any of the two following equations:

\[
d_H(A, B) = \max \left\{ \sup_{b \in B} d(b, A), \sup_{a \in A} d(a, B) \right\} = \sup_{x \in M} \inf_{c \in C} \rho(x, c)
\]

(1)

From a mathematical perspective, it is a basic and classical result that the Hausdorff distance is indeed a distance on the set of compact subsets of a metric space. In TDA, it can be a commodious way to quantify the proximity between varying data sets issued from the same ambient metric space.

2.2 Clustering

Using some specific rules to cluster the data set can extract details in a given data set. Time-series data sets are usually presented as finite metric spaces.

A finite metric space is defined by a pair \( (X, d(x)) \), where \( X \) is a finite set and \( d(x) : X \times X \rightarrow \mathbb{R} \) is a distance function which has been shown above. \( Q \) is the collection of all finite metric spaces and for any given \( n \in \mathbb{N} \), we use \([1 : n]\) to denote the set \{1, 2, ..., n\}. For instance, if \((X, d(x)) \in Q\), we denote the collection of all partitions of \( X \) by \( P(X) \). Each \( P \in P(X) \) can be treated as a block of \( P \). We denote by \( \mathcal{P} \), the collection of all pairs \( (X, P(X)) \) where \( X \in M \) and \( P(X) \in P(X) \). Formally,

\[
\mathcal{P} = \{(X, P) | X \in Q, P \in P(X)\}
\]

(2)

Definition 3 (Clustering Method). A clustering method \( \mathcal{C} \) is a map \( \mathcal{C} : \mathcal{M} \rightarrow \mathcal{P} \) such that for every \((X, d(x)) \in \mathcal{M}, \mathcal{C}((X, d)) = (X, P(X)) \), where \( P(X) \in P(X) \). Clustering represents partitioning \( n \) observations into \( k \) clusters. And a cluster is characterized with the concept of homogeneity, the similarity of observations within a cluster, and separation, which is the dissimilarity of observations from different clusters.

2.3 Simplicial Complexes and Persistent Homology

Definition 4 (Filtered Simplicial Complex). Consider a sequence of simplicial complexes \( \{K_\delta\}_{\delta \in \mathbb{R}^+} \) and for all \( \delta \leq \delta', K_\delta \subseteq K_{\delta} \). This sequence is a filtered simplicial complex.

Definition 5 (Rips Complex). The Rips complex \( R_\psi(r) \) is the simplicial complex defined as

\[
R_\psi(r) = \{ \sigma \subset X : d(x_i, x_j) < 2r, \forall x_i, x_j \in \sigma \}
\]

(3)

In order to figure the topological feature of the data set, persistent homology is an efficient way. A filtration \( G \) is a collection of sub-spaces closing to the data points at different resolutions. Bottleneck distance is a metric which can measure the distance between persistent diagram \( A \) and diagram \( B \).

Definition 6 (Bottleneck Distance). The bottleneck distance between persistent homology of the filtration \( PH_k(f) \) and \( PH_k(g) \) is defined by where the set \( \Gamma \) consists of all the bijections \( \gamma : Dgm_k(f) \cup \text{Diag} \rightarrow Dgm_k(g) \cup \text{Diag} \), and \( \text{Diag} \) is the diagonal \( \{(x, x) : x \in \mathbb{R} \} \subset \mathbb{R}^2 \) with infinite multiplicity.

\[
d_B(PH_k(f), PH_k(g)) = \inf_{\gamma \in \Gamma} \sup_{p \in Dgm_k(f)} ||p - \gamma(p)||_\infty
\]

(4)

2.4 Persistent Landscapes and Silhouettes

Definition 7 (Persistent Landscapes). Consider a gather of continuous, piece-wise linear functions
\( \lambda : \mathbb{N} \times \mathbb{R} \to \mathbb{R} \) that summarizes a persistent diagram and define the functions as the persistent landscape. See Figure 1. An example of persistence landscape (right) associated to a persistence diagram (left). The first landscape is in blue, the second one in red and the last one in orange. All the other landscapes are zero. For details about the conception. The idea of landscape is defined by considering the set of functions created by tenting each point \( p = (d, b) \) representing a birth-death pair in the persistent diagram as follows:

\[
\Lambda_p(t) = \begin{cases} 
    t-d, & t \in \left[ d, \frac{d+b}{2} \right] \\
    n-t, & t \in \left( \frac{d+b}{2}, n \right] \\
    0, & \text{otherwise}
\end{cases}
\]  

The persistent landscape of \( \text{dgm} \) is the collection of functions \( \{ \Lambda_p \}_p \):

\[
\lambda_{\text{dgm}}(k, t) = \max_p \Lambda_p(t), t \in [0, T], k \in \mathbb{N}
\]  

\( \text{K}_{\text{max}} \) is the \( k \)th largest value in the set. Particularly, given \( k \in \mathbb{N} \), the function \( \lambda_{\text{dgm}}(k, .) : \mathbb{R} \to \mathbb{R} \) is called the \( k \)-th landscape of \( \text{dgm} \). The map which connect to each persistent diagram is injective. In another word, all of the information will be saved well when a persistent diagram is presented through its persistent landscape.

![Figure 1](image.png)

**Figure 1.** An example of persistence landscape (right) associated to a persistence diagram (left). The first landscape is in blue, the second one in red and the last one in orange. All the other landscapes are zero.

Definition 8 (Silhouette). Let \( \text{Dgm}_k(\mathcal{F}) \) be a persistent diagram which incorporates \( N \) off diagonal birth-death pairs \( \{(b_j, d_j)\}_{j=1}^N \). The silhouette is defined as the following power-weighted function,

\[
\phi(q)(t) = \frac{\sum_{j=1}^N |d_j-b_j|^q \Lambda_j(t)}{\sum_{j=1}^N |d_j-b_j|^q}
\]  

for \( 0 < q < \infty \). In fact, when \( q \to \infty \), silhouettes converge to the first order landscapes \((k = 1)\).

3. Example: Analysis of financial data

In this section, the article uses persistent landscapes and their norms \((p = 1 \text{ and } p = 2)\) to process financial time series. It can be known that these two norms show massive advance preferred to financial crashes, while they show a docile behaviour when the market is steady. The research will first explain the method of analysis. Then the experiment will test it on some synthetic time series. Finally, there are four time series datasets downloaded from Yahoo Finance will be analysed.

3.1 A brief description of the method
Consider series \( \{x^k_n\}_{n} \), where \( k = 1, \ldots, m \) with \( m \) times, and a sliding window with the size \( v \). For every instance \( t_n \), we can have a point \( x(t_n) = (x^1_n, \ldots, x^m_n) \in \mathbb{R}^m \). After that, for every window with the size \( v \) we will have a point cloud data set made of \( v \) points in \( \mathbb{R}^m \), viz.

\[
X_n = (x(t_n), x(t_{n+1}), \ldots, x(t_{n+v-1}))
\]

Analyzing from the algorithmic perspective, what we can find is that the above point cloud is essentially a \( m \times v \) matrix, where \( m \) is the number of columns that corresponds to the number of 1D time series under study and \( w \) is the size of the window, which identifies the length of a column. After computing the persistent diagram of the Rips filtration, the corresponding persistent landscape, and \( L^p \)-norms of each point, we use the R-package named "TDA" to help the implementation, which highly simplifies the application programming interface.

### 3.2 Analysis of the financial data

Then we figure out the log-returns of the mentioned time series data. The log-returns are defined as the forward daily changes of each index as well as each trading day. In the log of the ratio \( r_{ij} = \ln \left( \frac{P_{i,j}}{P_{i-1,j}} \right) \), \( P_{i,j} \) stands for the adjusted ending value of the index \( j \) at the day \( i \). Investigating the topological characteristics of this multidimensional time series is the final target. In this example, we use \( m \) points in \( \mathbb{R}^d \) to form the point cloud. The coordinates of each point can figure the daily log-returns. We choose two size of the window: \( m = 60 \) and \( m = 100 \) trading days. Figure illustrates the Rips persistent diagrams and the corresponding 1-st landscape function \( \lambda_1 \). And \( \lambda_1 \) is defined:

\[
\lambda_k(x) = k - \max\{f_{(b_a,d_a)}(x) | (b_a, d_a) \in P_k\}
\]

In some data sets, we calculate the sliding window for 60 days. The \( x \) coordinate of Rips persistent diagrams corresponds to the times of birth \( (m) \) and the \( y \) coordinate stands for the death \( (n) \). The \( x \) and \( y \) coordinate correspond to the new axes: \( x = (m + n)/2 \), and \( y = (m - n)/2 \). Figure shows a externally distinct interval of topological signals from a topological noise per point cloud. What can be easily seen from the figure is that as the stock market becomes more explosive loops, the point clouds get much more obvious.
To measure this behavior, we calculate the $L^p$ norms ($p = 1, 2$) of the persistent landscapes $\lambda$ for each rolling window. The set of these values forms the daily time series of these measures, we can see fig. 3 for an example.

The newness of this method:
(i) Using the unitary sliding window to inclose the point cloud by the number of time series under consideration;
(ii) Using the $L^p$-norm of persistent landscapes to improve the stability of topological features.

**Figure 2.** The solid black dots represent connected components, red triangles represent loops.

**Figure 3.** Calculate the time series of normalized persistence landscapes with the sliding window of 60 days. ($L1$: blue line and $L2$: red line)
4. Further applications to other domains and Discussion

4.1 Mapper Algorithm

The Mapper algorithm was advanced by Singh, Mmoli and Carlsson, as a geometrical tool to visualize and analyze data sets. The intuitive idea behind Mapper is stated in fig. 4 and can be explained as follows:

Suppose there is a point cloud data representing a shape, for example a “knot”. First the experiment project the whole data on a coordinate system with less dimensionality in order to reduce complexity via dimensionality reduction (here project the data on the knot to x-axis). Now we partition the parameter space (x-axis) into several bins with an overlapping percentage. Next, put data into overlapping bins. Afterwards, we use clustering algorithms in order to classify the points of each bin into several clusters. Once the previous stage is done, interactive graph can be created. Implementation of mapper algorithm has already been available in python packages like “Mapper” and “Kepler Mapper”.

Figure 4. Mapper algorithm on knot shape data cloud: a) First we project the whole data cloud to embedded space (here x-axis). b) Then we partition the embedded space into overlapping bins (here showed as colored intervals). c) Then we put data into overlapping bins. d) Next we use any clustering algorithm to cluster the points in the cloud data. e and f) each cluster of points in every bin would represented as a node of the graph and we draw and edge between two nodes if they share a common data point.

4.2 Applications to Biology

This section shows some applications of TDA to bioscience. In [2] the essay proposed an approach to automatically categorize neuronal network dynamics. The main approach is using topological features of spaces built from spike train distances. The authors in paper [7] use the concept of zigzag persistent homology [5] to make clear the possibility of forgetting information in the model for memory. And the results show that the rodent needs to balance the relationship between remembering and forgetting. In [17], persistent homology is used for characterize, identify and classify protein firstly. The authors extracted molecular topological fingerprints[10], TDA is used to characterize the structure of chromatin. In a diverse way, another topological method for studying finite metric spaces is Mapper [14].

4.3 Applications to Other Domains

Persistent homology has also played pivotal role in shape classification. The authors of [6] used P-H to identify signatures of finite metric spaces. The authors then sample points from every shape to form a finite metric space, compare the spaces, and finally adapt this approach to compare diverse shapes. In[12], P-H is used to establish a descriptor for identifying and comparing zeolites. The authors performed high-throughput screening which is based on descriptor and identified best zeolites for
pollution gas capture applications. The results concluded in [12] correspond to the existing results on top-performing zeolites for these applications.

4.4 Persistent homology and machine learning

In some domains, after obtain diagrams from data sets, the diag can be straight translated and exploited for better understanding of the information. However, persistent homology are not always doing well in all kinds of data sets. In other words, the non-linear diagrams prevent them from being directly used as normal features in some machine learning algorithms. In this circumstance, persistence landscapes and the variants which have been presented in Section 2.4 afford a primarily choice to translate persistence diagrams to elements of a vector space. Varieties of vector summaries of persistence diagrams have been proposed and used for different data sets. For instance, basic conclusion of the applications are showed in [4]. The author extracted Betty curves from persistence diagrams and then used the curves with 1-dimensional Convolutional Neural Networks (CNN) to analyze time dependent data.

Conclusions and Future Work

This article introduces the basic concepts and the main mathematics used for TDA, uses persistent landscapes to process financial time series and discusses the further applications of TDA. Through the experiment and discussion shown above, it can be concluded that Topological data analysis is an active and growing area of current research. The main goal of this article is using TDA to Process time-series Data in a Persistent Homology way. However, other perspectives from TDA are still deserved more exploration. Therefore, further studies will be done in the future in other areas using diverse ways.

References

[1] Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P., Chepushtanova, S., Hanson, E., Motta, F. and Ziegelmeyer, L. (2017) Persistence images: A stable vector representation of persistent homology. J. The Journal of Machine Learning Research, 18(1):218-252.
[2] Bardin, J., Spreemann, G. and Hess, K. (2019) Topological exploration of artificial neuronal network dynamics. J. Network Neuroscience, 3(3):725-743.
[3] Bendich, P., Marron, J., Miller, E., Pieloch, A. and Skwerer, S. (2016) Persistent homology analysis of brain artery trees. J. The annals of applied statistics, 10(1):198.
[4] Bonis, T., Ovsjanikov, M., Oudot, S. and Chazal, F. (2016) Persistence-based pooling for shape pose recognition. In: International Workshop on Computational Topology in Image Context, Marseille, pp. 19-29.
[5] Carlsson, G., Silva, V. and Morozov, D. (2009) Zigzag persistent homology and real-valued functions. In: Proceedings of the twenty-fifth annual symposium on Computational geometry, Aarhus, pp. 247-256. ACM.
[6] Chazal, F., Cohen-Steiner, D., Guibas, L., M´emoli, F. and Oudot, S. (2009) Gromov-hausdorff stable signatures for shapes using persistence. J. Computer Graphics Forum, 28:1393-1403.
[7] Samir Chowdhury, Bowen Dai, and Facundo M´emoli. The importance of forgetting: Limiting memory improves recovery of topological characteristics from neural data. PloS one, 13(9):e0202561, 2018.
[8] Dabaghian, Y., M´emoli, F., Frank, L. and Carlsson, G. (2012) A topological paradigm for hippocampal spatial map formation using persistent homology. J. PLoS computational biology, 8(8):e1002581.
[9] De Cecco, L., Nicolau, M., Giannoccaro, M., Daidone, M., Bossi, P., Locati, L., Licitra, L. and Canevari, S. (2015) Head and neck cancer subtypes with biological and clinical relevance: Metaanalysis of gene-expression data. J. Oncotarget, 6(11):9627.
[10] Emmett, K., Schweinhart, B. and Rabadan, R. (2015) Multiscale topology of chromatin folding. arXiv preprint arXiv:1511.01426.
[11] Gidea, M. and Katz, Y. (2018) Topological data analysis of financial time series: Landscapes of crashes. J. Physica A: Statistical Mechanics and its Applications, 491:820-834.

[12] Lee, Y., Barthel, S., Dlotko, P., Moosavi, S., Hess, K. and Smit, B. (2018) High-throughput screening approach for nanoporous materials genome using topological data analysis: application to zeolites. J. Journal of chemical theory and computation, 14(8):4427-4437.

[13] Obayashi, I., Hiraoka, Y. and Kimura, M. (2018) Persistence diagrams with linear machine learning models. J. Journal of Applied and Computational Topology, 1(3-4):421-449.

[14] Singh, G., M’emoli, F. and Carlsson, G. (2007) Topological methods for the analysis of high dimensional data sets and 3d object recognition. In: Symposium on Point Based Graphics, Prague, pp. 91-100.

[15] Topaz, C., Ziegelmeier, L. and Halverson, T. (2015) Topological data analysis of biological aggregation models. PloS one, 10(5):e0126383.

[16] Umeda, Y. (2017) Time series classification via topological data analysis. J. Information and Media Technologies, 12:228-239.

[17] Xia K. and Wei, G. (2014) Persistent homology analysis of protein structure, flexibility, and folding. J. International journal for numerical methods in biomedical engineering, 30(8):814-844, 2014.