Time Series Featurization via Topological Data Analysis: an Application to Cryptocurrency Trend Forecasting

Kwangho Kim *, Jisu Kim †, Alessandro Rinaldo ‡

December 10, 2018

Abstract

We propose a novel methodology for feature extraction from time series data based on topological data analysis. The proposed procedure applies a dimensionality reduction technique via principal component analysis to the point cloud of the Takens’ embedding from the observed time series and then evaluates the persistence landscape and silhouettes based on the corresponding Rips complex. We define a new notion of Rips distance function that is especially suited for persistence homologies built on Rips complexes and prove stability theorems for it. We use these results to demonstrate in turn some stability properties of the topological features extracted using our procedure with respect to additive noise and sampling. We further apply our method to the problem of trend forecasting for cryptocurrency prices, where we manage to achieve significantly lower error rates than more standard, non TDA-based methodologies in complex pattern classification tasks. We expect our method to provide a new insight on feature engineering for granular, noisy time series data.

Keywords: stability, topological features, Rips distance function, noisy time series, time series featurization, topological data analysis, cryptocurrency forecasting

*{Department of Statistics & Data Science, Machine Learning Department}, Carnegie Mellon University, Pittsburgh, PA 15213. Email: kwhangkk@stat.cmu.edu

†Inria Saclay – Île-de-France, Palaiseau, France. Email: jisu.kim@inria.fr, URL: http://www.stat.cmu.edu/~jisuk/

‡Department of Statistics & Data Science, Carnegie Mellon University, Pittsburgh, USA. Email: arinaldo@cmu.edu, URL: http://www.stat.cmu.edu/~arinaldo/
Time series analysis aims at predicting the evolution of a time varying phenomenon. Time series data exhibit different characteristics than point cloud data obtained through random sampling. In the first place, data points close in time are more likely to share common features than those which are further apart in time; this is referred to as serial autocorrelation. Moreover in time series forecasting we rarely know the underlying data generating distribution and the stochastic dependence across time associated to it, making applications of standard asymptotic results, such as the central limit theorem and the weak law of large numbers problematic. Lastly, the granularity of the features are often very different from that of the outcome variable so simple summary statistics do not carry nuanced information about time-varying patterns.

These issues render the bulk of modern machine learning and feature engineering methods based on i.i.d. data inadequate, or at the very least cause a degradation in their performance. Practitioners instead rely on stochastic models that have been developed based on the Ergodic theorem (e.g. ARIMA, SARIMA) or time-frequency analysis techniques (e.g. Fourier or Wavelet transform). More recently, due to the tremendous increase in the computing processing power and the developments of novel methodologies and theories for data analysis, researchers have started harnessing more innovational approaches once considered irrelevant to time series analysis (e.g., deep-learning architectures [35, 2, 38, 26]). One such approach that has attracted growing interest is topological data analysis.

Topological data analysis (TDA) is a recent and emerging field of data science that rely on topological and geometric tools to infer relevant features for possibly complex data [15]. TDA has provided alternative procedures to the traditional statistical and machine learning-based algorithms. TDA methods are now applied in many areas, ranging from document structure representation [65] to cancer detection [40], task-optimal architecture of neural networks [27], to name a few (see [15] for more examples). TDA, especially persistent homology, has gained increasing importance for the analysis of noisy signals and time series across many disparate domains. For example, TDA used in time delay embedding has been deployed for problems in signal processing and systems engineering [15, 14, 59, 51, 1, 23, 16, 61, 63, 21]. The use of persistent homology also appears in a series of recent contributions in financial econometrics [25, 11]. We refer the interested readers to [24] for a more comprehensive review of recent applications of TDA in time series analysis.

In particular, one may hope that TDA methods could succeed when traditional time series methods and machine learning algorithms seemingly fail to provide adequate predictive and descriptive solutions in problems involving complex time series. One such instance is the analysis of the cryptocurrency market. Traditional asset-pricing and risk-modeling theories are not applicable to the cryptocurrencies at the current time given their embryonic stage. Consequently, practitioners have ended up relying on a purely data-driven approach for cryptocurrency predictions. Despite a series of recent attempts, results have been mixed due to data noisiness, sudden wild swings in sentiment, and sampling period errors [1, 34, 50]. A few research have reported significant improvements in predictive performance when they use deep learning models [47, 38, 26]. However, the granularity of data used in their analysis is limited to only daily basis and the model performance is mostly measured by mean-squared-error, whereas practitioners are usually more interested in classifying more complex patterns spanned on multiple time points.

In this paper, we propose a novel time series featurization method to construct informative and robust features based on concepts and methods from TDA. We will provide a framework where one can successfully harness topological features embedded in the dynamical system
for an observed time series, and the theoretical guarantees that our featurization is robust
to sampling noise. In particular, our method provides an effective solution to feature engi-
neering for granular time series data where a rich set of inherent patterns cannot be easily
captured by simple summary statistics or conventional models. After extracting the fea-
tures, we can use flexible machine learning tools to perform prediction or classification. We
apply the proposed method to fine-grained bitcoin price dataset (up to minute-by-minute
basis) for complex pattern classifications over multiple time points (e.g. the occurrence of
a rare peak in next $k$ hours). We evaluate the performance of our method and compare it
to other alternative models.

Contributions: We make the following contributions. (1) We propose a denoising process
via principal component analysis after the state-space reconstruction. (2) We derive features
based on homological persistence by directly vectorizing persistent landscape or silhouette
[13]. (3) We provide a novel stability theorem that guarantees the robustness of our TDA
features to sampling noise by introducing a new notion of Rips distance function. (4) We
demonstrate the value of the proposed framework through bitcoin price forecasting where
we perform not only a single time-point prediction but also complex pattern classifications.

1 TOPOLOGICAL DATA ANALYSIS

In this section we review basic concepts about persistence homology and stability theorems.
We refer the reader to Appendix C and [28, 20, 10, 17] for further details and formal
definitions. Throughout, $X$ will denote a subset of $\mathbb{R}^d$ and $\mathcal{X}$ a collection of points from it.
We use the notation $B_{\mathbb{R}^d}(x, r)$ for the open ball in $\mathbb{R}^d$ centered at $x$ with radius $r$, and for
a set $A \subset \mathbb{R}^d$, let $\text{conv}(A)$ be the convex hull of $A$.

1.1 The Rips Complex

When inferring topological properties of $X$ from $\mathcal{X}$, we rely on simplicial complexes, discrete
structures built over the observed points to provide a topological approximation of the
underlying space. Due to computational issues, the most common choice is the Vietoris-
Rips complex, also referred to as Rips complex, where simplexes are built based on pairwise
distances among its vertices.

Definition 1. The Rips complex $R_\mathcal{X}(r)$ is the simplicial complex defined as

$$R_\mathcal{X}(r) := \{ \sigma \subset \mathcal{X} : d(x_i, x_j) < 2r, \forall x_i, x_j \in \sigma \}. \quad (1)$$

1.2 Persistent Homology

Persistent homology is a multiscale approach to represent topological features. A filtration
$\mathcal{F}$ is a collection of subspaces approximating the data points at different resolutions.

Definition 2. A filtration $\mathcal{F} = \{ \mathcal{F}_a \}_{a \in \mathbb{R}}$ is a collection of sets with $a \leq b$ implying
$\mathcal{F}_a \subset \mathcal{F}_b$.

For a filtration $\mathcal{F}$ and for each $k \in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$, the associated persistent homology $H_k \mathcal{F}$
is a collection of $k$-th dimensional homology of each subset in $\mathcal{F}$.
Definition 3. Let $\mathcal{F}$ be a filtration and let $k \in \mathbb{N}_0$. The associated $k$-th persistent homology $PH_k \mathcal{F}$ is a collection of vector spaces $\{H_k F_a\}_{a \in \mathbb{R}}$ equipped with homomorphisms $\{i_k^{a,b}\}_{a \leq b}$, where $H_k F_a$ is the $k$-th dimensional homology of $F_a$ and $i_k^{a,b} : H_k F_a \to H_k F_b$ is the homomorphism induced by the inclusion $F_a \subset F_b$.

For the $k$-th persistent homology $PH_k \mathcal{F}$, the set of filtration levels at which a specific homology appears is always an interval $[b, d) \subset [-\infty, \infty]$, i.e. a specific homology is formed at some filtration value $b$ and dies when the inside hole is filled at another value $d > b$.

Definition 4. Let $\mathcal{F}$ be a filtration and let $k \in \mathbb{N}_0$. The corresponding $k$-th persistence diagram $Dgm_k(\mathcal{F})$ is a finite multiset of $(\mathbb{R} \cup \{\infty\})^2$, consisting of all pairs $(b, d)$ where $[b, d)$ is the interval of filtration values for which a specific homology appears in $PH_k \mathcal{F}$. $b$ is called a birth time and $d$ is called a death time.

1.3 Landscapes and Silhouettes

The space of persistence diagrams is a multiset, which makes it difficult to analyze and feed as input to learning or statistical methods. Hence it is useful to transform the persistent homology into a functional Hilbert space, where the analysis is easier and learning methods can be directly applied. Examples include persistent landscapes and silhouettes \[12, 13, 5\].

We first define a set of functions $t \in \mathbb{R}^+ \mapsto \Lambda_p(t)$ for each birth-death pair $p = (b, d)$ in the persistence diagram as follows:

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

(2)

For each birth-death pair $p$, $\Lambda_p(\cdot)$ is piecewise linear. Then the persistence landscape $\lambda$ of the diagram is the overlay of the functions $\{\Lambda_p\}_p$ defined as

$$
\lambda(k, t) = k \max_p \Lambda_p(t), \quad t \in [0, T], k \in \mathbb{N},
$$

(3)

where $k \max$ is the $k$-th largest value in the set. Usually we set $k = 1$.

To define a silhouette, consider $N$ off diagonal birth-death points $\{(b_j, d_j)\}_{j=1}^N$. Then we define the power-weighted silhouette as, for $0 < q < \infty$,

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

(4)

The value $q$ can be thought of as controlling the trade-off parameter between uniformly treating all pairs in the persistence diagram and considering only the most persistent pairs. In fact when $q \to \infty$, silhouettes converge to the first order landscapes ($k = 1$).

1.4 Stability Theorems

Stability theorems have been established when the filtration $\mathcal{F}$ is generated from the sub-level sets or the super-level sets of a function. Let $f, g : X \to \mathbb{R}$ be two functions, and
let $PH_*(f)$ and $PH_*(g)$ be the corresponding persistent homologies of the sublevel set filtrations \( \{ f \leq L \}_{L \in \mathbb{R}} \) and \( \{ g \leq L \}_{L \in \mathbb{R}} \).

To impose stability, we first endow the space of persistence diagrams with a metric. The most fundamental one is the bottleneck distance.

**Definition 5.** The bottleneck distance between the persistent homology of the filtrations $PH_*(f)$ and $PH_*(g)$ is defined by

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

where the set $\Gamma$ consists of all the bijections $\gamma : Dgm_k(f) \cup Diag \to Dgm_k(g) \cup Diag$, and $Diag$ is the diagonal \( \{(x,x) : x \in \mathbb{R}\} \subset \mathbb{R}^2 \) with infinite multiplicity.

When two functions $f$ and $g$ satisfy a regular condition called tameness (see Appendix C), their bottleneck distance is bounded by their $\ell_\infty$ distance, an important and useful fact known as the stability theorem.

**Theorem 6 (Stability Theorem).** [18, 10, 17] For two tame functions $f, g : X \to \mathbb{R}$,

$$d_B(PH_k(f), PH_k(g)) \leq \| f - g \|_\infty.$$

## 2 RIPS DISTANCE FUNCTION AND CONSISTENCY OF THE RIPS COMPLEX

When utilizing the Rips complex for inferring topological information of the underlying space from the observed points as in Section 1.1, it is important to identify the information extractable from the Rips complex. Such information will be particularly useful if it is stable with respect to small changes in the underlying space. Also from a statistical standpoint, it is desirable to consistently estimate such information as the number of data points grows.

In other words, we are trying to answer the following three questions:

1. What topological properties of $X$ can be extracted from computing the Rips complex?
2. Is the information robust to changes of the underlying space?
3. Can we estimate such properties arbitrarily well, using the Rips complex, as we collect more and more data?

To answer the first question we propose a new notion of **Rips distance function**, which is a distance function adapted to the Rips complex. A classical approach to extract topological information from a subset of the metric space $X$ is to consider a sublevel set of the distance function, measuring the distance from any point in $X$ to that subset. To adapt to Rips complexes, we define the Rips distance function as follows.

**Definition 7 (Rips distance function).** For a set $X \subset \mathbb{R}^d$, let the Rips distance function $d_R^X : \mathbb{R}^d \to \mathbb{R}$ be

$$d_R^X(x) = \inf \{ r + s : \exists x_1, \ldots, x_n \in X, \exists y_i \in \mathbb{R}^d(x_i, s) \text{ with } \| y_i - y_j \|_2 < 2r \forall i, j \leq n, x \in \text{conv}(\{x_1, \ldots, x_n\}) \}$$
See Appendix \[\square\] for more properties of the Rips distance function defined above. The Rips distance function corresponds to the Rips complex in terms of the persistent homology. For a set \( X \), we write \( PH_*(d^R_X) \) for the persistent homology of the sublevel filtration of the Rips distance function \( \{(d^R_X)^{-1}(-\infty, r)\}_{r \in \mathbb{R}} \), and \( PH_*(R_X) \) for the persistent homology of the Rips filtration \( \{R_X(r)\}_{r \in \mathbb{R}} \). Then two persistent homologies are equal:

**Proposition 2.1.** For a finite set \( X \subset \mathbb{R}^d \), two persistent homologies \( PH_*(d^R_X) \) and \( PH_*(R_X) \) coincide, i.e. \( PH_*(d^R_X) = PH_*(R_X) \).

Now we turn to the second question, which is the stability of the Rips distance function with respect to the underlying space. We answer this by the following stability theorem, where the bottleneck distance of persistent homologies of two Rips distance functions is bounded by the corresponding Gromov-Hausdorff distance (see Definition \[\square\] in Appendix). Hence, if the set \( X \) is perturbed by a small amount, the corresponding Rips distance function also changes a little.

**Theorem 8** (Stability of the Rips distance function). Let \( X, Y \subset \mathbb{R}^d \) be two bounded sets and \( d^R_X, d^R_Y \) be the corresponding Rips distance functions. Then the bottleneck distance between \( PH_*(d^R_X) \) and \( PH_*(d^R_Y) \) is upper bounded by the Gromov-Hausdorff distance as

\[
d_B(PH_*(d^R_X), PH_*(d^R_Y)) \leq d_{GH}(X, Y).
\]

Finally, we turn to the third question, which is about the consistency of the Rips complex with respect to the Rips distance function. We consider two cases, where the data can be either fixed or random. Combining Proposition 2.1 and Theorem 8 gives that the bottleneck distance between the Rips filtration and the Rips distance function is upper bounded by the Gromov-Hausdorff distance. Hence, for fixed data when the observed data is sufficiently dense, then the topological information of the Rips distance function of the underlying space is closely approximated by the Rips complex on the observed points in \( X \).

**Corollary 2.1** (Consistency for the fixed data). Let \( X \) be a bounded set, and \( X \subset X \) be a finite subset. Then the bottleneck distance between \( PH_*(R_X) \) and \( PH_*(d^R_X) \) is upper bounded by the Gromov-Hausdorff distance as

\[
d_B(PH_*(R_X), PH_*(d^R_X)) \leq d_{GH}(X, X).
\]

Now, we consider the random data, where we observe i.i.d. data from a probability distribution supported on the underlying space. Then as the number of points goes to \( \infty \), the bottleneck distance between the Rips filtration of the observed points and the Rips distance function of the underlying space goes to 0 almost surely. This implies that as we sample more points, it is more likely infer the topological information of the underlying space via the Rips distance function from the Rips complex formed on the observed points.

**Theorem 9** (Consistency for the random data). Let \( P \) be a distribution on \( \mathbb{R}^d \) with \( \text{supp}(P) \) being bounded. Let \( X = \{X_1, \ldots, X_n\} \) be i.i.d. from \( P \). Then

\[
d_B(PH_*(R_X), PH_*(d^R_{\text{supp}(P)})) \to 0 \text{ a.s.}
\]
3 METHODOLOGY

3.1 Dynamical Systems and Takens Embedding

A time series can be considered a series of projections of the observed states from a dynamical system which is a rule for time evolution on a state space. We typically reconstruct the state space of the dynamical system and its transition rules through attractors [57, 43, 19] (see Appendix B for more information).

However, generating an attractor directly from observed data is in theory impossible. This is basically due to the fact that an attractor is comprised of infinitely many points whereas we have only finitely many observed time series data. One well-known alternative is to form a quasi-attractor based on Takens’ delay embedding theorem [55]. Let $M^0$ denote the manifold corresponding to the original dynamical system generating the observed time series data. Takens’ delay embedding theorem guarantees the existence of a smooth map $\Psi$ such that $M^0 \rightarrow M'$, where $M'$ is an $m$-dimensional Euclidean space and $M^0$ and $M'$ are topologically equivalent. [49] found that this is guaranteed provided that $m > d_0$, where $d_0$ is the box-counting dimension of the attractor in $M^0$. [48] generalized Takens’ theorem to infinite-dimensional systems that have finite-dimensional attractors.

Hence, once we obtain $\Psi$ we can instead study $M'$ to analyze the underlying mechanism of given time series. This result has been popularized since it gives the ability to extract crucial information about time series of interest in Euclidean space. This embedding theorem has been applied to a wide range of scientific disciplines to study chaotic or noisy time series [58, 9, 30, 3, 56, 33, 31, 54]. In particular, it can be used for state-space reconstruction of the original dynamical system so that TDA is applicable to point cloud data (e.g., [61, 21, 63, 60]).

We adopt time-delayed sliding window embedding following the work of [45, 59]. However, we have modified their approach to be more suitable to our framework and, in particular, to more easily verify any stability properties of our TDA features (section 4).

For now, we only consider one dimensional time series data. Let $f(t)$ be a function defined over the non-negative reals $\mathbb{R}_{+0}$. Since time series featurization will be done only upon finitely bounded time interval, it is sufficient to consider a function $f(t)$ on an interval $t \in [0, T]$ with $0 < T < \infty$. First let

$$x = \{x_1, x_2, ..., x_N\} = \{f(\delta T), f(2\delta T), ..., f(T)\},$$

be an observed sequence which consists of $N$ time-ordered sampled points from $f$ uniformly over $[0, T]$ where $\delta T = T/N$. Then, let the sliding window mapping $SW_{m,\tau} f : R \rightarrow R^m$ be

$$SW_{m,\tau} f(t) := [f(t - (m - 1)\tau), ..., f(t - \tau), f(t)]^\top.$$

In other words, for a fixed function $f$ the map $t \mapsto SW_{m,\tau} f(t)$ generates the $m$ most recent equally-spaced points up to time $t$ with a gap $\tau$. In fact, parameters that appear in above definition characterize the transformation according to Takens’ embedding theorem; $\tau$ will be the delay parameter and $m$ will be our embedding dimension (i.e. $\text{dim}(M')$). Now we
construct the trajectory matrix $X \in \mathbb{R}^{\{N-(m-1)\tau\} \times m}$ as follows:

$$X = \begin{bmatrix}
SW_{m,\tau} f(\delta_T(1 + (m - 1)\tau))^T \\
SW_{m,\tau} f(\delta_T(2 + (m - 1)\tau))^T \\
\vdots \\
SW_{m,\tau} f(T)^T
\end{bmatrix}$$

By computing (5) we obtain the continuous transformation from the original manifold $\mathcal{M}_0$ to $\mathcal{M}' \in \mathbb{R}^m$. By Takens’ theorem (and its generalized version given in [48]) there always exists a pair $(\tau, m)$ such that the vectors generated via the transformation described in (5) are on a manifold topologically equivalent to $\mathcal{M}_0$. In fact, the last display in (5) is a typical expression for Takens’ delay embedding transformation [63, 21, 46, 51]. The pattern traced out by $X$ in $\mathbb{R}^m$ is referred to as the quasi-attractor in that it shares some topological equivalence with the original attractor in $\mathcal{M}_0$ (for formal definition, see, for example, [58]).

The transformation procedure described above requires the choice of the embedding dimension, $m$, and the time delay, $\tau$. There is no generic optimal method for choosing these parameters without any structural assumptions [52, 53], and practitioners rather rely on heuristic methods. $m$ must be sufficiently large to satisfy $m > 2d_0$ but we never know $d_0$ in reality. To determine $d_0$, we count linearly independent vectors in the singular value decomposition of $X$ following the work of [44, 58], and pick a set of consecutive numbers equal or greater than that. As for $\tau$, this parameter can be set using autocorrelation or mutual information [52]. Nonetheless $\tau$ is more often chosen in a qualitative manner. In the simulation, we use a grid of values for $(m, \tau)$ and report the average.

3.2 Feature Extraction with Persistent Homology

In the previous section, we have shown how we can extract information about the underlying dynamical system of given time series by analyzing the quasi-attractor from the point cloud generated via the trajectory matrix $X$ in (5). In this section, we develop a methodology in which we can translate the valuable information about the quasi-attractor into vector-valued features using persistent homology methods.

TDA is known to be very useful to analyze point cloud data [8]. As seen in the previous section we can find a reasonable transformation from the original time series to its point cloud representation which is topologically equivalent to the dynamical system of interest. From the point cloud that we have constructed, we form the Rips complex and the corresponding compute persistent homology. It is usually hard to directly use persistence diagrams as features since the diagram itself is very sparse and vectorization is also not trivial. Hence we propose to use instead vectorized version of the persistent landscapes and silhouettes as our final TDA features.

Even though there are several applications where researchers find harnessing persistent homology quite effective in pattern recognition or signal detection, only a few of them
explicitly describe specific procedures of constructing time series features and provide a rationale behind it. For example, [45, 59] focus on quantifying periodicity in particular types of synthetic signals. Studies such as [25, 51, 4, 60, 23] have suggested that persistent homology can lead to a new type of features from time series or signal data, but their work mostly came away with a form of simple visualizations. [46] used persistent homology to extract time series features for spatial clustering but their featurization only rely on the partial information they subjectively choose.

Our proposed method resembles some of prior works [61, 63, 21] in that we rely on the delayed embedding and compute its persistent homology to featurize topological properties. However, all of the aforementioned methods use Betti numbers as their TDA features, which are straightforward to vectorize. More importantly, the theoretical underpinning for these methods is very limited. In this paper, we instead directly vectorize persistent landscapes or silhouettes for our final TDA features and we verify, via stability theorems, that they show favorable theoretical properties.

Before computing the persistent homology, we perform a denoising step to reduce insignificant noise-like homological features via principal component analysis (PCA). When true signal is lying on a low-dimensional subspace while the error stretches to all dimensions, which is common in practice, PCA has an effect of reducing the homological noise aligned orthogonally to the signal space while preserving the homological features of the original signal. See Figure 1 for simple examples. We thus perform PCA to transform $X$ into $X^l$ to bring down the dimension from $\mathbb{R}^m$ to $\mathbb{R}^l$ where $l \ll m$. Next, we construct the Vietoris-Rips filtration $R_{X^l}^l = \{R_{X^l}(r)\}_{r \in \mathbb{R}}$ and compute its persistence diagram. As we show below, our algorithm and analysis explicitly take into consideration this extra denoising process, which is another contribution of this paper. Other dimensionality reduction techniques could be considered in place of PCA, such as LLE, ISOMAP, Kernel PCA or diffusion maps, which may better adapt to the intrinsic geometric of the data.

### 3.3 Algorithm

In this section, we present our algorithm implementing the proposed featurization method in a step-by-step manner. First, we form a point cloud $X$ in $\mathbb{R}^m$ from the observed time series data via (5) with our choice of $(m, \tau)$. After the point cloud embedding, we perform PCA on $X$ to reduce the dimension from $\mathbb{R}^m$ to $\mathbb{R}^l$, and obtain $X^l$. Next, we compute the persistence diagram $D_{X^l}$ from the Rips complex $R_{X^l}$, and obtain the landscapes or silhouette functions accordingly. Finally, we vectorize each of the landscapes or silhouette functions so that we can use them as inputs of machine learning algorithms. For vectorization, we set a resolution parameter $\kappa$ and then take $\lceil r_{max}/\kappa \rceil$ evenly spaced samples from each landscape or silhouette function. If our maximum dimension is $d_{max}$, then in the end we will have a vector of length $(d_{max} + 1)\lceil r_{max}/\kappa \rceil$ as our final TDA feature. We detail entire procedure in Algorithm 1.
Figure 1: An example of case in which dimensionality reduction by PCA reduces the homological noise – assumed for convenience to be aligned orthogonally to the true signal space – while preserving the homological features from the signal. We focus on the 1-dimensional feature, marked as red triangles on the right column. The signal consists of 10 equally spaced points on a unit circle (top left). This has one feature in the persistence diagram (top right). Noise is added to the tangential direction to the circle and z-axis to sample 100 points (mid left). Then several noisy features appear near the diagonal line in the persistence diagram (mid right). After doing PCA to project to 2-dimensional space (bottom left), all the noisy features are gone but the signal feature is preserved in the persistence diagram (bottom right).
Algorithm 1. Time Series Featurization via TDA

**Input:** Time series sequence $x = \{x_1, x_2, \ldots, x_N\}$

1. Construct the point cloud $X \in \mathbb{R}^m$ via the trajectory matrix (5) with $m, \tau$
2. Perform PCA on $X$ and obtain $X^l \in \mathbb{R}^l$ ($l \ll m$)
3. Construct the Rips complex $R_{X^l}$ and compute the persistence diagram $D_{X^l}$ with parameters $d_{\text{max}}, r_{\text{max}}$
4. From $D_{X^l}$, compute the landscape $\lambda(k, t)$ from (3) or the silhouette $\phi^{(p)}(t)$ from (4)
5. Vectorize $\lambda(k, t)$ or $\phi^{(p)}(t)$ at resolution $\kappa$

**Output:** Vector of length $(d_{\text{max}} + 1)[r_{\text{max}}/\kappa]$

Note that our algorithm is basically for one dimensional time series with complex patterns. One naive extension to multivariate time series might be simply concatenating all the TDA features, where each of which has been obtained from different time series, then perform dimensionality reduction. It is worth noting that as the dimension of the homological features increases (i.e. for large $k$) both landscape and silhouette have only few spikes and are mostly zeros elsewhere. In other words, our features become very sparse. Hence even if we incorporate multiple TDA features from various covariates into our analysis, which potentially renders our problem very high-dimensional, we can still perform predictive analysis assuming sparsity. It has been shown that there are machine learning methods which attain acceptable convergence rate with high-dimensional data under sparsity assumption even in nonparametric setting (for example, see [36, 39]).

We expect our method to be more effective with granular data. The more granular our data, the more patterns to exploit. Moreover, as discussed in the following section, the proposed featurization method is robust to sampling noise. Consequently, the proposed algorithm will likely be good at identifying/featurizing many recurring transition rules and patterns that are relevant to predicting target signal yet buried in noisy granular data.

4 STABILITY ANALYSIS

Now we show that our method is robust against noise and finite sampling errors. Suppose there is a true signal function $f : [0, T] \rightarrow \mathbb{R}$ which is corrupted with additive noise $\epsilon : [0, T] \rightarrow \mathbb{R}$, and we end up observing $N$ sample points $x$ from $f + \epsilon$ uniformly over $[0, T]$. Our goal is to infer topological features of the Takens embedding of $f$ only using $x$.

In what follows, we provide a novel stability theorem which guarantees that the persistence diagram of the Rips filtration from the described method does not arbitrary get farther away from the persistence diagram of our target.

**Proposition 4.1.** Let $f : [0, T] \rightarrow \mathbb{R}$ be a Lipschitz function with Lipschitz constant $L_f$. Suppose $x$ is a time-series of $N$-samples from $f + \epsilon$ within the time interval $[0, T]$. Let $X := \{SW_{m, \tau} f(t) : t \in [0, T]\}$ be the Takens embedding of $f$. Let $X$ be a point cloud constructed via (5) with fixed $m \in \mathbb{R}$. Let $X^l$ be the outcome of the PCA on $X$. Suppose
\[
\frac{X^\top X}{N-(m-1)\tilde{r}} \text{ has } \tilde{l} \leq l \text{ positive eigenvalues } \lambda_1 \geq \ldots \geq \lambda_l > 0. \text{ Let } D_X, D_f \text{ denote the persistence diagrams of } \{R_X(t)\}_{t \in \mathbb{R}} \text{ and } \{d^R_X(-\infty, r)\}_{r \in \mathbb{R}}, \text{ respectively. Then,}
\[
d_B(D_X, D_f) \leq \sqrt{m}\|\epsilon\|_\infty + \frac{m^3 L_f T \|\epsilon\|_\infty (4L_f T + \|\epsilon\|_\infty)}{\lambda_l} + \frac{\sqrt{mL_f T}}{N}. \tag{6}
\]
From above, we obtain an analogous stability result for the landscapes and silhouettes, both of which are the final output of our method.

**Theorem 10.** Let \( D_X, D_f \) denote persistence diagrams from Proposition 4.1. Let \( \lambda_{k,X}, \lambda_{k,f} \) be the landscape functions from \( D_X \) and \( D_f \). Then
\[
\|\lambda_{k,X} - \lambda_{k,f}\|_\infty \leq \sqrt{m}\|\epsilon\|_\infty + \frac{m^3 L_f T \|\epsilon\|_\infty (4L_f T + \|\epsilon\|_\infty)}{\lambda_l} + \frac{\sqrt{mL_f T}}{N}. \tag{7}
\]
Also, let \( \phi_X^{(p)}, \phi_f^{(p)} \) be the silhouette functions from \( D_X \) and \( D_f \). Then
\[
\|\phi_X^{(p)} - \phi_f^{(p)}\|_\infty \leq \sqrt{m}\|\epsilon\|_\infty + \frac{m^3 L_f T \|\epsilon\|_\infty (4L_f T + \|\epsilon\|_\infty)}{\lambda_l} + \frac{\sqrt{mL_f T}}{N}. \tag{8}
\]
For the bounding term of the stability theorems in (6), (7), (8), the first and second terms are errors from the noise \( \epsilon \) and the PCA process respectively, and are proportional to \( \|\epsilon\|_\infty \) or \( \|\epsilon\|_\infty^2 \). The last term is due to observing discrete time domain, and is inversely proportional to \( N \). Hence, when the noise \( \epsilon \) is small and the size of our data is large enough, then we are guaranteed to closely approximate the topological information of the signal from our method.

## 5 ILLUSTRATIVE SIMULATIONS

This section provides synthetic examples showing how the proposed topological features returned by Algorithm 1 successfully reflect crucial topological information about original time series. Example commonly shown in previous studies mostly have been limited to learning simple geometrical features (i.e. circles or spheres; for example, see \([45, 59, 60]\)). Unlike most of the previous works, which have provided only deterministic examples for illustrations of persistent homology, here we consider instead more complex stochastic models. To this end, we employ three types of stochastic model as follows.

- **ARIMA\(_{1,1,2}\):** 
  \[x_t = x_{t-1} + \phi_1 \Delta x_{t-1} - \theta_1 e_{t-1} - \theta_2 e_{t-2}\]
- **Composite Sinusoidal:** 
  \[asin(x_t)sin(bx_t) + c \cos(x_t + d)\]
- **Ornstein-Uhlenbeck:** 
  \[dx_t = \theta(\mu - x_t)dt + \sigma dW_t\]

Here \( W_t \) denotes the Wiener process and we use \((\phi_1, \theta_1, \theta_2) = (0.4, 0.2, 0.1), a, b, c, d \sim \text{Uniform}[1.25, 1.75] \) and \((\theta, \mu, \sigma) = (-0.5, 0, 0.5)\). The above processes generate times series data for damped-trend linear exponential smoothing, composite sinusoidal functions with different frequencies, and mean-reversion random process respectively. We repeat a simulation three times for each model, generating 500 samples each, and consequently have
nine distinct time series sequences in total. Then we extract topological features from each sampled sequences using Algorithm 1. We set $m = 25, \tau = 5$, and use silhouettes with $p = 0, 5, 1, 1$ for dimension $0, 1, 2$ respectively. To compute the persistence diagram we use the R package TDA [22].

Figure 2 shows graphical representations of topological features of the nine synthetic times series data over different homological dimensions. Surprisingly, different models tend to have different topological features that are summarized as a set of peaks across different dimensions. Through this synthetic experiment, we gain an important insight into the possibility that our featurization method is able to distinguish complex mechanisms underlying time series data.

![Figure 2](image)

Figure 2: Original time series data (left) and their topological features represented as silhouette function (right). Different line types and colors correspond to different models and simulations, respectively.

6 APPLICATION TO CRYPTOCURRENCY DATA

As mentioned in the introduction, the analysis of cryptocurrency data is challenging. The embryonic nature of the technology, unreliable sentiment tracking, inadequate pricing and forecasting methodologies, and suspected price manipulation have created an incredibly complex and noisy dataset that is difficult to predict. It has very little systematic risk, which implies the cryptocurrency price itself should be the most valuable source of information about its price movement. Moreover, a study of [50] has intimated that using more granular data would be more beneficial. These data characteristics offer a tremendously fertile ground for TDA to be effective in price prediction for cryptocurrency, rather than other traditional financial assets in which various asset pricing models are already available.

We prepare Bitcoin price data during the period from 01/01/2014 to 07/31/2018, at three different sampling rates - daily, hourly, and minute-by-minute basis. We use the return over a single period $r_t = (p_t - p_{t-1})/p_{t-1}$ based on two consecutive closing prices $p_t$ and $p_{t-1}$. 
6.1 Setup

In this section we detail our simulation schemes and baseline methods for comparison. For the simulation, we set $y$ as a recurring pattern of our interest over single or multiple time period. It can be a simple price change in the very next time point (e.g. whether price goes up/down) or more complex patterns over multiple time points (e.g. whether a rare price jump/drop occurs in next six hours). We assume that there exists a set of time-series signals $\{f_1, f_2, \ldots\}$ where each signal is highly likely to occur somehow prior to the predetermined pattern $y$ (see Figure 3). Note that $y$ is known by user yet $f_j$’s are unknown. Moreover in reality we only observe discrete samples from each $f_j$ with a potentially significant amount of noise. Our purpose is to effectively identify and featurize those $f_j$’s, $j = 1, 2, \ldots$, using Algorithm 1, where we can utilize those features to build a model to predict a likelihood of occurrence of the pattern $y$.

![Figure 3: An example of recurring pattern $y$ and time-series signals $f_1, f_2$ that are very similar in shape and highly likely to occur prior to $y$. In reality we only observe discrete samples from each $f_1, f_2$ with potentially significant amount of noise. Data are obtained from minute-by-minute observations in first 3 hours on 07/01/2018.](image)

We let $N_{\text{training}}$ and $N_{\text{test}}$ denote the number of samples in training and test period respectively. To apply Algorithm 1, we use a sequence of length $N \ll N_{\text{training}}$ to generate TDA features within the training period. Then we fit a machine learning model that predicts the predetermined pattern $y$ from the TDA features using samples in the training period, and use the trained model to get predicted $\hat{y}$ from samples in the test period. Then we compute a test error rate and roll over into next training, test window (rolled over by $N_{\text{test}}$). Throughout the simulation, we use $N_{\text{training}} = 336$, $N_{\text{test}} = 24$, and $N = 168$.

We carry out two types of simulation. The first one is a prediction where $y \in \mathbb{R}$, and we simply predict $\hat{y}_{t+1}$ pointwise based on the information available up to time $t$. The second one is a classification where $y$ is a categorized pattern over next $k$ time points $[0, k], k \in \mathbb{N}$. We use four patterns, each of which yields either 3-class or binary classification problem, defined as follows. **P1**: price movement in the next (single) time point with three regimes of $\{\text{up, down, neutral}\}$. **P2**: determine if a rare jump (defined by $(1 - \alpha)$-quantile) occurs in next $k$ time points. **P3**: determine if a rare drop (defined by $\alpha$-quantile) occurs in next $k$ time points. **P4**: price movement on average in next $k$ time points with three regimes of $\{\text{up, down, neutral}\}$. Figure 4 illustrates an example of each pattern. For simulation we use $k = 6$ and $\alpha = 0.1$. When we define a regime in **P1**, **P4** we use thresholds to have almost the same number of samples in each class.

We construct TDA features via Algorithm 1. In the simulation, we try various combinations...
Figure 4: Examples of how the four patterns have their values. In the figure, we classify both $P_1$ and $P_4$ as up, and a rare jump/drop occurs in $P_2$, $P_3$ respectively.

of $(m, \tau)$ and report the average rate while we restrict the number of principal components to three ($l = 3$). Then we form an ensemble of widely used nonparametric models to fit the best prediction/classification model. Specifically, we use cross-validation-based superlearner ensemble algorithm \cite{62} via the SuperLearner package in R to combine support vector machine, random forest, k-nearest neighbor regression, and multivariate adaptive regression splines (NP1). We also use AdaBoost decision tree algorithm (NP2) via the fastAdaboost package in R as our second nonparametric ensemble method. We use default methods (mostly cross-validation) in the package to tune any hyperparameters in the models.

As mentioned previously, we believe that a recent price history itself contains the most valuable information about the short-term price movement. Hence first we proceed univariate analysis where we obtain TDA features only from the bitcoin price data. Next we include an additional covariate to see whether it increases the accuracy of our predictive analysis. Following \cite{7}, we use trading volume as the additional covariate. In both cases, we use three different data granularities (sampling rates): daily (D), hourly (H), and every minute (M).

We employ five baseline methods to compare the performance. First, we include two widely used time series models ARIMA and ARIMAX, where the latter incorporates an external regressor into ARIMA model. When there is no additional covariate we use first $N$ coefficients of Fourier transform as the external regressor. Next, following studies of \cite{26, 38} we use deep learning architecture; we use single layer convolutional neural network (CNN) where we insert a convolutional layer of stride 12 and use entire $N$ time points as input. Finally, we use more naive, traditional featurization method (NAIVE); we compute the sample moments up to fourth order, three quartile values, minimum and maximum value from each of length $N$ sequence and concatenate them.

6.2 Results

When we produce TDA features, we vary the value of $\tau, m$ within the range $m \in \{24, ..., 48\}$, and $\tau \in \{3, 4, 5, 6\}$ and only report the average results. Figure 5 summarizes the first simulation results where we predict a single value $\hat{y}_{t+1}$ based on the information available up to $t$. We notice that utilizing TDA features results in better performance in overall. However, more drastic improvement in prediction accuracy appears in the second simulation where we classify future price movements over multiple timepoints into predetermined categories. As we observe in Figure 6 while all the baseline methods have barely improved in performance from the base rate, the two nonparametric classifiers based on TDA features show
significant improvements from the base rate. In both results, the contribution in prediction accuracy from the trading volume has turned out to be marginal. As we expected, the more granular our dataset is, the more informative TDA features appear to be in general. Most remarkably, benefits from utilizing TDA features are more evident in complex pattern classifications than single time point forecasts. We include Table 1 and Table 2 in Appendix A for more detailed information about the simulation results.

Figure 5: Normalized RMSE ($\times 10^{-2}$) of the proposed and baseline methods. On x-axis, Uni/S, Co/S mean univariate analysis at granularity level S, analysis with covariate at granularity level S respectively, where $S \in \{D, H, M\}$.

Figure 6: Classification error rate (the proportion of misclassified observations in test set) of the proposed and baseline methods. We present result for univariate analysis (left) and analysis with covariate (right) separately. In both figures, on x-axis $N/S$ means classification for pattern $P_N$ at granularity $S$, where $S \in \{D, H, M\}$ and $P_N \in \{P1, P2, P3, P4\}$.
7 CONCLUSION

In this study we have proposed a novel time series featurization method based on TDA, which can be potentially effective in identifying recurring patterns that are unknown but relevant to the target signal of interest. We implement an extra denoising process after the state-space reconstruction and provide a novel stability theorem by introducing a new notion of Rips distance function. We verify the effectiveness of our method by applying it to cryptocurrency trend forecasting, where we achieved improved error rates particularly in complex pattern classification.
References

[1] Jethin Abraham, Daniel Higdon, John Nelson, and Juan Ibarra. Cryptocurrency price prediction using tweet volumes and sentiment analysis. *SMU Data Science Review*, 1(3):1, 2018.

[2] Ratnadip Adhikari and RK Agrawal. An introductory study on time series modeling and forecasting. *arXiv preprint arXiv:1302.6613*, 2013.

[3] Arslan Basharat and Mubarak Shah. Time series prediction by chaotic modeling of nonlinear dynamical systems. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1941–1948. IEEE, 2009.

[4] Kenneth A Brown and Kevin P Knudson. Nonlinear statistics of human speech data. *International Journal of Bifurcation and Chaos*, 19(07):2307–2319, 2009.

[5] Peter Bubenik. Statistical topological data analysis using persistence landscapes. *The Journal of Machine Learning Research*, 16(1):77–102, 2015.

[6] Dmitri Burago, Yuri Burago, and Sergei Ivanov. *A course in metric geometry*, volume 33 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2001. URL: https://doi.org/10.1090/gsm/033

[7] John Y Campbell, Sanford J Grossman, and Jiang Wang. Trading volume and serial correlation in stock returns. *The Quarterly Journal of Economics*, 108(4):905–939, 1993.

[8] Gunnar Carlsson. Topological pattern recognition for point cloud data. *Acta Numerica*, 23:289–368, 2014.

[9] Martin Casdagli. Nonlinear prediction of chaotic time series. *Physica D: Nonlinear Phenomena*, 35(3):335–356, 1989.

[10] Frédéric Chazal, David Cohen-Steiner, Marc Glisse, Leonidas J Guibas, and Steve Y Oudot. Proximity of persistence modules and their diagrams. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*, pages 237–246. ACM, 2009.

[11] Frédéric Chazal, Vin de Silva, and Steve Oudot. Persistence stability for geometric complexes. *Geometriae Dedicata*, 173(1):193–214, 2014. URL: https://doi.org/10.1007/s10711-013-9937-z

[12] Frédéric Chazal, Brittany Fasy, Fabrizio Lecci, Bertrand Michel, Alessandro Rinaldo, and Larry Wasserman. Subsampling methods for persistent homology. In *International Conference on Machine Learning*, pages 2143–2151, 2015.

[13] Frédéric Chazal, Brittany Terese Fasy, Fabrizio Lecci, Alessandro Rinaldo, and Larry Wasserman. Stochastic convergence of persistence landscapes and silhouettes. In *Proceedings of the thirtieth annual symposium on Computational geometry*, page 474. ACM, 2014.

[14] Frédéric Chazal, Leonidas J Guibas, Steve Y Oudot, and Primoz Skraba. Persistence-based clustering in riemannian manifolds. *Journal of the ACM (JACM)*, 60(6):41, 2013.
[15] Frédéric Chazal and Bertrand Michel. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *arXiv preprint arXiv:1710.04019*, 2017.

[16] Frédéric Chazal and Steve Oudot. Towards persistence-based reconstruction in euclidean spaces. *CoRR*, abs/0712.2638, 2007. URL: [http://arxiv.org/abs/0712.2638](http://arxiv.org/abs/0712.2638), arXiv:0712.2638.

[17] Frédéric Chazal, Steve Y. Oudot, Marc Glisse, and Vin De Silva. *The Structure and Stability of Persistence Modules*. SpringerBriefs in Mathematics. Springer Verlag, 2016. URL: [https://hal.inria.fr/hal-01330678](https://hal.inria.fr/hal-01330678).

[18] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. volume 37, pages 103–120, Jan 2007. URL: [https://doi.org/10.1007/s00454-006-1276-5](https://doi.org/10.1007/s00454-006-1276-5), doi:10.1007/s00454-006-1276-5.

[19] J-P Eckmann and David Ruelle. Fundamental limitations for estimating dimensions and lyapunov exponents in dynamical systems. *Physica D: Nonlinear Phenomena*, 56(2-3):185–187, 1992.

[20] Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Soc., 2010.

[21] Saba Emrani, Thanos Gentimis, and Hamid Krim. Persistent homology of delay embeddings and its application to wheeze detection. *IEEE Signal Processing Letters*, 21(4):459–463, 2014.

[22] Brittany Terese Fasy, Jisu Kim, Fabrizio Lecci, and Clément Maria. Introduction to the r package tda. *arXiv preprint arXiv:1411.1830*, 2014.

[23] Jennifer Gamble and Giseon Heo. Exploring uses of persistent homology for statistical analysis of landmark-based shape data. *Journal of Multivariate Analysis*, 101(9):2184–2199, 2010.

[24] Shafie Gholizadeh and Wlodek Zadrozny. A short survey of topological data analysis in time series and systems analysis. *arXiv preprint arXiv:1809.10745*, 2018.

[25] Marian Gidea and Yuri Katz. Topological data analysis of financial time series: Landscapes of crashes. *Physica A: Statistical Mechanics and its Applications*, 491:820–834, 2018.

[26] Sneha Gullapalli. Learning to predict cryptocurrency price using artificial neural network models of time series. 2018.

[27] William H Guss and Ruslan Salakhutdinov. On characterizing the capacity of neural networks using algebraic topology. *arXiv preprint arXiv:1802.04443*, 2018.

[28] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[29] Bruce Henry, Nigel Lovell, and Fernando Camacho. Nonlinear dynamics time series analysis. *Nonlinear biomedical signal processing: Dynamic analysis and modeling*, 2:1–39, 2001.
[30] Yan Huang and Gang Kou. Manifold learning-based phase space reconstruction for financial time series. In PACIS, page 364, 2014.

[31] AW Jayawardena and Feizhou Lai. Analysis and prediction of chaos in rainfall and stream flow time series. Journal of hydrology, 153(1-4):23–52, 1994.

[32] Holger Kantz and Thomas Schreiber. Nonlinear time series analysis, volume 7. Cambridge university press, 2004.

[33] Eric J Kostelich and Thomas Schreiber. Noise reduction in chaotic time-series data: A survey of common methods. Physical Review E, 48(3):1752, 1993.

[34] Connor Lamon, Eric Nielsen, and Eric Redondo. Cryptocurrency price prediction using news and social media sentiment. Not Published, 2016.

[35] Martin Längkvist, Lars Karlsson, and Amy Loutfi. A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognition Letters, 42:11–24, 2014.

[36] Han Liu. Nonparametric Learning in High Dimensions. PhD thesis, Carnegie Mellon University, School of Computer Science, Machine Learning Department, 2010.

[37] Zonghua Liu. Chaotic time series analysis. Mathematical Problems in Engineering, 2010, 2010.

[38] Brandon Ly, Divendra Timaul, Aleksandr Lukanan, Jeron Lau, and Erik Steinmetz. Applying deep learning to better predict cryptocurrency trends. 2018.

[39] Lukas Meier, Sara Van de Geer, Peter Bühlmann, et al. High-dimensional additive modeling. The Annals of Statistics, 37(6B):3779–3821, 2009.

[40] Vidit Nanda and Radmila Sazdanović. Simplicial models and topological inference in biological systems. In Discrete and topological models in molecular biology, pages 109–141. Springer, 2014.

[41] Ashadun Nobi, Sungmin Lee, Doo Hwan Kim, and Jae Woo Lee. Correlation and network topologies in global and local stock indices. Physics Letters A, 378(34):2482–2489, 2014.

[42] Tohru Ozaki. 2 non-linear time series models and dynamical systems. Handbook of statistics, 5:25–83, 1985.

[43] Norman H Packard, James P Crutchfield, J Doyne Farmer, and Robert S Shaw. Geometry from a time series. Physical review letters, 45(9):712, 1980.

[44] Milan Palu and I Dvoak. Singular-value decomposition in attractor reconstruction: pitfalls and precautions. Physica. D, Nonlinear phenomena, 55(1-2):221–234, 1992.

[45] Jose A Perea and John Harer. Sliding windows and persistence: An application of topological methods to signal analysis. Foundations of Computational Mathematics, 15(3):799–838, 2015.

[46] Cásio MM Pereira and Rodrigo F de Mello. Persistent homology for time series and spatial data clustering. Expert Systems with Applications, 42(15-16):6026–6038, 2015.
[47] Jonathan Rebane, Isak Karlsson, Stojan Denic, and Panagiotis Papapetrou. Seq2seq
rnnss and arima models for cryptocurrency prediction: A comparative study. 2018.

[48] James C Robinson. A topological delay embedding theorem for infinite-dimensional
dynamical systems. *Nonlinearity*, 18(5):2135, 2005.

[49] Tim Sauer, James A Yorke, and Martin Casdagli. Embedology. *Journal of statistical
Physics*, 65(3-4):579–616, 1991.

[50] Gustav Segerstedt and Theodor Uhmeier. How accuracy of time-series prediction for
cryptocurrency pricing is affected by the sampling period, 2018.

[51] Lee M Seversky, Shelby Davis, and Matthew Berger. On time-series topological data
analysis: New data and opportunities. In *Proceedings of the IEEE Conference on
Computer Vision and Pattern Recognition Workshops*, pages 59–67, 2016.

[52] Michael Small. Optimal time delay embedding for nonlinear time series modeling.
*arXiv preprint nlin/0312011*, 2003.

[53] Fernanda Strozzi, José-Manuel Zaldivar, and Joseph P Zbilut. Application of nonlinear
time series analysis techniques to high-frequency currency exchange data. *Physica A:
Statistical Mechanics and its Applications*, 312(3-4):520–538, 2002.

[54] George Sugihara and Robert M May. Nonlinear forecasting as a way of distinguishing
chaos from measurement error in time series. *Nature*, 344(6268):734, 1990.

[55] Floris Takens. Detecting strange attractors in turbulence. In *Dynamical systems and
turbulence, Warwick 1980*, pages 366–381. Springer, 1981.

[56] James Theiler, Stephen Eubank, André Longtin, Bryan Galdrikian, and J Doyne
Farmer. Testing for nonlinearity in time series: the method of surrogate data. *Physica
D: Nonlinear Phenomena*, 58(1-4):77–94, 1992.

[57] Howell Tong. *Non-linear time series: a dynamical system approach*. Oxford University
Press, 1990.

[58] Thomas K Torku. Takens theorem with singular spectrum analysis applied to noisy
time series. 2016.

[59] Christopher J Tralie and Jose A Perea. (quasi) periodicity quantification in video data,
using topology. *SIAM Journal on Imaging Sciences*, 11(2):1049–1077, 2018.

[60] Patrick Truong. An exploration of topological properties of high-frequency one-
dimensional financial time series data using tda, 2017.

[61] Yuhei Umeda. Time series classification via topological data analysis. *Information and
Media Technologies*, 12:228–239, 2017.

[62] Mark J Van der Laan, Eric C Polley, and Alan E Hubbard. Super learner. *Statistical
applications in genetics and molecular biology*, 6(1), 2007.

[63] Vinay Venkataraman, Karthikeyan Natesan Ramamurthy, and Pavan Turaga. Persistent
homology of attractors for action recognition. In *Image Processing (ICIP), 2016
IEEE International Conference on*, pages 4150–4154. IEEE, 2016.
[64] Y. Yu, T. Wang, and R. J. Samworth. A useful variant of the Davis-Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015. URL: https://doi.org/10.1093/biomet/asv008, doi:10.1093/biomet/asv008.

[65] Xiaojin Zhu. Persistent homology: An introduction and a new text representation for natural language processing. In *IJCAI*, pages 1953–1959, 2013.
APPENDIX

A  Simulation Results

| Method           | Univariate | With Covariate |
|------------------|------------|----------------|
|                  | D | H | M | D | H | M |
| TDA + NP1        | 14.5 | 13.6 | 12.1 | 14.1 | 12.5 | 10.9 |
| TDA + NP2        | 15.5 | 13.4 | 13.9 | 13.3 | 11.8 |
| ARIMA            | 29.5 | 35.2 | 38.9 | - | - | - |
| ARIMAX           | 14.8 | 13.9 | 14.1 | 13.6 | 13.7 | 13.1 |
| CNN              | 17.3 | 16.7 | 14.6 | 16.5 | 16.8 | 15.0 |
| NAIVE + NP1      | 43.1 | 48.4 | 43.6 | 39.0 | 40.9 | 38.1 |
| NAIVE + NP2      | 49.1 | 44.9 | 42.6 | 41.8 | 39.9 | 37.7 |

Table 1: Normalized RMSE ($\times 10^{-2}$) of the proposed and baseline methods

| Method | Daily | Hourly | Every minute |
|--------|-------|--------|--------------|
|        | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $P_1$ | $P_2$ | $P_3$ | $P_4$ |
| TDA + NP1 | 0.55 | 0.35 | 0.36 | 0.57 | 0.48 | 0.25 | 0.29 | 0.51 | 0.43 | 0.22 | 0.25 | 0.44 |
| TDA + NP2 | 0.57 | 0.33 | 0.33 | 0.55 | 0.44 | 0.23 | 0.25 | 0.47 | 0.40 | 0.20 | 0.23 | 0.41 |
| ARIMA | 0.67 | 0.65 | 0.68 | 0.67 | 0.66 | 0.65 | 0.68 | 0.65 | 0.66 | 0.67 | 0.68 | 0.67 |
| ARIMAX | 0.63 | 0.62 | 0.63 | 0.65 | 0.59 | 0.57 | 0.64 | 0.62 | 0.55 | 0.56 | 0.63 |
| CNN | 0.60 | 0.57 | 0.61 | 0.55 | 0.61 | 0.56 | 0.63 | 0.60 | 0.54 | 0.55 | 0.60 |
| NAIVE + NP1 | 0.54 | 0.63 | 0.64 | 0.62 | 0.53 | 0.62 | 0.65 | 0.62 | 0.55 | 0.63 | 0.64 | 0.60 |
| NAIVE + NP2 | 0.52 | 0.62 | 0.63 | 0.60 | 0.63 | 0.59 | 0.51 | 0.62 | 0.65 | 0.64 | 0.60 |

Table 2: Classification error rate (the proportion of misclassified observations in test set) of the proposed and baseline methods. In each cell the lower number corresponds to the case that we incorporate additional covariate.

B  A Brief Review of Dynamical Systems

Dynamical systems are mathematical objects used to model phenomena with states that evolve over time, and widely used in various scientific domains including quantum physics. We describe such dynamical evolutions as a set of state transition rules typically represented by differential equations. A time series can be considered a series of projections of the
observed states from such a dynamical system, and therefore reconstructing the equations for the transition rules from the observed time series data is crucial to understand the underlying phenomena [43]. The problem is that it is in general very difficult to fully reconstruct the equations for the transition rules of the dynamical systems from observed time series data without any a priori knowledge [57, 19]. The most widely used approach to bypass this problem is to utilize an attractor. Putting it simply, an attractor is a set of numerical values toward which the given dynamical system eventually evolves over time, passing through various transitions determined by the system, without depending too much on its starting conditions. It is very common to have two time series signals that have the same transition rule but completely different observed wave forms. In contrast, the attractors of the dynamical system constructed from these time series still closely resemble each other. Thus studying the attractors provide a means to model dynamical systems (see, for example, [37, 29, 3, 42, 57, 32] for more details).

C More Background for Topological Data Analysis

This section introduces more detailed background in algebraic topology for topological data analysis that are used in this paper to supplement Section 1.

C.1 Distance between sets on metric spaces

When topological information of the underlying space is approximated by the observed points, it is often needed to compare two sets with respect to their metric structures. Here we present two distances on metric spaces, Hausdorff distance and Gromov-Hausdorff distance. We refer to [6] for more details and other distances.

The Hausdorff distance is on sets embedded on the same metric spaces. This distance measures how two sets are close to each other in the embedded metric space. When $S \subset X$, we denote by $U_r(S)$ the $r$-neighborhood of a set $S$ in a metric space, i.e. $U_r(S) = \bigcup_{x \in S} B_X(x, r)$.

**Definition 11** (Hausdorff distance). ([6, Definition 7.3.1])

Let $X$ be a metric space, and $X, Y \subset X$ be a subset. The Hausdorff distance between $X$ and $Y$, denoted by $d_H(X, Y)$, is defined as

$$d_H(X, Y) = \inf\{r > 0 : X \subset U_r(Y) \text{ and } Y \subset U_r(X)\}.$$ 

The Gromov-Hausdorff distance measures how two sets are far from being isometric to each other. To define the distance, we first define a relation between two sets called correspondence.

**Definition 12.** Let $X$ and $Y$ be two sets. A correspondence between $X$ and $Y$ is a set $C \subset X \times Y$ whose projections to both $X$ and $Y$ are both surjective, i.e. for every $x \in X$, there exists $y \in Y$ such that $(x, y) \in C$, and for every $y \in Y$, there exists $x \in X$ with $(x, y) \in C$.

For a correspondence, we define its distortion by how the metric structures of two sets differ by the correspondence.
Definition 13. Let $X$ and $Y$ be two metric spaces, and $C$ be a correspondence between $X$ and $Y$. The distortion of $C$ is defined by

$$\text{dis}(C) = \sup \left\{ |d_X(x,x') - d_Y(y,y')| : (x,y), (x',y') \in C \right\}.$$ 

Now the Gromov-Hausdorff distance is defined as the smallest possible distortion between two sets.

Definition 14 (Gromov-Hausdorff distance). ([6, Theorem 7.3.25])

Let $X$ and $Y$ be two metric spaces. The Gromov-Hausdorff distance between $X$ and $Y$, denoted as $d_{GH}(X,Y)$, is defined as

$$d_{GH}(X,Y) = \frac{1}{2} \inf_C \text{dis}(C),$$

where the infimum is over all correspondences between $X$ and $Y$.

C.2 Simplicial complex and Nerve Theorem

A simplicial complex can be seen as a high dimensional generalization of a graph. Given a set $V$, an (abstract) simplicial complex is a set $K$ of finite subsets of $V$ such that $\alpha \in K$ and $\beta \subset \alpha$ implies $\beta \in K$. Each set $\alpha \in K$ is called its simplex. The dimension of a simplex $\alpha$ is $\dim \alpha = \text{card} \alpha - 1$, and the dimension of the simplicial complex is the maximum dimension of any of its simplices. Note that a simplicial complex of dimension 1 is a graph.

When approximating the topology of the underlying space by observed samples, a common choice other than the Rips complex is the Čech complex, defined next. Below, for any $x \in X$ and $r > 0$, we let $B_X(x,r)$ denote the closed ball centered at $x$ and radius $r > 0$.

Definition 15 (Čech complex). Let $X \subset \mathbb{X}$ be finite and $r > 0$. The (weighted) Čech complex is the simplicial complex

$$\check{C}ech_X^n(r) := \{ \sigma \subset X : \cap_{x \in \sigma} B_X(x,r) \neq \emptyset \},$$

The superscript $\mathbb{X}$ will be dropped when understood from the context.

Note that the Čech complex and Rips complex have following interleaving inclusion relationship

$$\check{C}ech_X^n(r) \subset R_X^n(r) \subset \check{C}ech_X^n(2r).$$

In particular, when $\mathbb{X}$ is a Euclidean space, then the constant 2 can be tightened to $\sqrt{2}$:

$$\check{C}ech_X^n(r) \subset R_X^n(r) \subset \check{C}ech_X^n(\sqrt{2}r).$$

The topology of the Čech complex is linked to underlying continuous spaces via Nerve Theorem. Let $r > 0$ and consider the union of balls $\bigcup_{x \in \mathcal{X}} B_X(x,r)$. Then the union of balls is homotopic equivalent to the Čech complex by the following Nerve Theorem.

Theorem 16 (Nerve Theorem). Let $\mathcal{X} \subset \mathbb{X}$ be a finite set and $r > 0$. Suppose for any finite subset $\{x_1, \ldots, x_k\} \subset \mathcal{X}$, the intersection $\bigcap_{j=1}^k B_X(x_j,r)$ is either empty or contractible, then the Čech complex $\check{C}ech_{\mathcal{X}}^\mathbb{X}(r)$ is homotopic equivalent to the union of balls $\bigcup_{x \in \mathcal{X}} B_X(x,r)$.
C.3 Stability Theorem on Persistence Module

This section gives an introduction to the Stability Theorem on the persistence module. We refer to [17, 10] for more details.

A persistence module is an algebraic abstraction of a persistent homology.

**Definition 17 (Persistence module).** ([17, Section 2.1], [10, Definition 2.1]) A persistence module \( F \) is a family \( \{ F_L \}_{L \in \mathbb{R}} \) of vector spaces indexed by the elements of \( \mathbb{R} \), together with a family \( \{ f_r : F_r \to F_{r'} \}_{r \leq r'} \) of homomorphisms such that: \( \forall r \leq r' \leq r'' \), \( f_{r''} = f_{r'} \circ f_{r} \) and \( f_r = id_{F_r} \).

For the stability theorem to hold, we require local finiteness of the persistence module. We impose the images of the homomorphisms to be of finite rank, which we refer to as the tameness condition.

**Definition 18 (tameness).** ([17, Section 3.8])

We say that \( F \) is tame if the image \( \text{im}(f_{r'}) \) of the homomorphism \( f_{r'} : F_r \to F_{r'} \) is of finite rank for all \( r < r' \).

Now, we introduce the concept of interleaving.

**Definition 19.** Two persistence modules \( F \) and \( G \) are said to be \( \epsilon \)-interleaved if there exist two families of homomorphisms \( \{ \phi_r : F_r \to G_{r+\epsilon} \}_{L \in \mathbb{R}} \) and \( \{ \psi_r : G_r \to F_{r+\epsilon} \}_{r \in \mathbb{R}} \) such that the following diagrams commute for all \( r \leq r' \):

\[
\begin{align*}
F_r - \epsilon & \xrightarrow{\phi_{r-\epsilon}} F_{r'} + \epsilon \\
& \xrightarrow{\psi_r} G_r \\
& \xrightarrow{\psi_{r'}} G_{r'}
\end{align*}
\]

If two persistence modules are interleaved, then their bottleneck distance are close, which is the stability theorem.

**Theorem 20 (Stability Theorem).** ([17, Theorem 5.23], [10, Theorem 4.4]) Let \( F \) and \( G \) be two tame persistence modules. If \( F \) and \( G_r \) are \( \epsilon \)-interleaved, then \( d_B(F, G) \leq \epsilon \).

D Properties of the Rips distance function

In general, topological properties are well carried out via continuous functions. We first show that the Rips distance function is 1-Lipschitz, and hence continuous.

**Proposition D.1.** The Rips distance function of a set \( X \subset \mathbb{R}^d \) is 1-Lipschitz, i.e. for any \( x, y \in X \),

\[
|d^R_X(x) - d^R_X(y)| \leq \|x - y\|.
\]
The Rips distance function corresponds to the Rips complex through its sublevel sets. To see this, we first describe the sublevel set of the Rips distance function as follows.

**Lemma 21.** Fix a set $X \subset \mathbb{R}^d$ and $r > 0$. The sublevel set of the Rips distance function at $r$ can be described as

$$(d_{R \mathbb{R}^d}X)^{-1}(-\infty, r) := \left\{ z \in \mathbb{R}^d : \exists s, t \text{ with } s + t < r, \exists x_1, \ldots, x_n \in X, \exists y_i \in B_{\mathbb{R}^d}(x_i, s) \text{ with } \|y_i - y_j\| < 2t \forall i, j \leq n, z \in \text{conv}\{y_1, \ldots, y_n\} \right\},$$

When the base set $X$ is finite, the sublevel set of the Rips distance function corresponds to the Rips complex via homotopy and homology. This is analogous to the Nerve Theorem (Theorem 16), which shows the homotopic relation between the sublevel set of the usual distance function and the Čech complex (see Appendix C).

In fact, we show a stronger interleaving relation between the sublevel filtration of the Rips distance function and the Rips filtration, which will show both Proposition D.3 and Proposition 2.1.

**Proposition D.2.** Fix a finite set $X \subset \mathbb{R}^d$, and consider the sublevel filtration of the Rips distance function $\{(d_{R \mathbb{R}^d}X)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ and the Rips filtration $\{R_X(r)\}_{r \in \mathbb{R}}$. Then for all $r \in \mathbb{R}$, there exist simplicial maps $\phi_r : (d_{R \mathbb{R}^d}X)^{-1}(-\infty, r) \to R_X(r)$ and $\psi_r : R_X(r) \to (d_{R \mathbb{R}^d}X)^{-1}(-\infty, r)$ that commutes with the inclusion map on the homotopy or homology level, i.e. when $r \leq r'$, the following diagram commutes:

$$
\begin{array}{ccc}
(d_{R \mathbb{R}^d}X)^{-1}(-\infty, r) & \longrightarrow & (d_{R \mathbb{R}^d}X)^{-1}(-\infty, r') \\
\downarrow \phi_r & & \uparrow \psi_r \\
R_X(r) & \longrightarrow & R_X(r')
\end{array}
$$

Then the sublevel set of the Rips distance function corresponds to the Rips complex via homotopy and homology as a corollary.

**Proposition D.3.** Fix a finite set $X \subset \mathbb{R}^d$ and $r > 0$, the sublevel set of $d_{R \mathbb{R}^d}X$ at $r$, i.e. $(d_{R \mathbb{R}^d}X)^{-1}(-\infty, r)$ and the Rips complex at $r$, i.e. $R_X(r)$ are homotopic, and their homologies coincide.

Now, we observe that the Rips distance function is stable as a function with respect to the underlying space.

**Proposition D.4.** Let $X, Y \subset \mathbb{R}^d$ be two subsets and $d_{R \mathbb{R}^d}X, d_{R \mathbb{R}^d}Y$ be the corresponding Rips distance functions. Then

$$\|d_{R \mathbb{R}^d}X - d_{R \mathbb{R}^d}Y\|_\infty \leq d_H(X, Y).$$

When we show the stability theorem for the persistent homology of sublevel sets of a function, a key assumption is that the sublevel filtration is tame. This is a standard regular assumptions in topological data analysis. For the Rips distance function, a sufficient condition is that the underlying space is bounded.

**Proposition D.5.** Let $X \subset \mathbb{R}^d$ be a bounded set and $d_{R \mathbb{R}^d}X$ be its Rips distance function. Then the sublevel filtration $\{(d_{R \mathbb{R}^d}X)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ is tame.
Proposition D.5 shows another benefit of using the Rips distance function. For Rips complex, tameness is guaranteed for only finite points, but the Rips distance function can guarantee for any bounded points as well. In general, the Rips distance function has a simpler filtration structure, so guaranteeing regularity conditions will be easier.

For the Stability Theorem of the persistent homology of the Rips distance function in Theorem 8, the bounded conditions on the sets are partly for the tameness condition. Once we weakened to the tameness condition, then we can drop some of the bounded conditions.

First, when we use the Hausdorff distance instead of the Gromov-Hausdorff distance for bounding the bottleneck distance, then we can drop both bounded conditions on sets. In other words, the bottleneck distance between persistent homologies of Rips distance functions is bounded by the Hausdorff distance of the underlying sets with only tameness conditions.

**Proposition D.6.** Let $X, Y \subset \mathbb{R}^d$ be two subsets of $\mathbb{R}^d$, and let $d_X^R$, $d_Y^R$ be the corresponding Rips distance functions of $X$ and $Y$, respectively. Suppose two sublevel filtrations $\{(d_X^R)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ and $\{(d_Y^R)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ are tame. Then the bottleneck distance $d_B(\text{PH}_s(d_X^R), \text{PH}_s(d_Y^R))$ is bounded by the Hausdorff distance between $X$ and $Y$, i.e.

$$d_B(\text{PH}_s(d_X^R), \text{PH}_s(d_Y^R)) \leq d_H(X, Y).$$

Second, when we use the Gromov-Hausdorff distance for bounding the bottleneck distance, we need one of the sets to be bounded but not necessarily both. In other words, the bottleneck distance between persistent homologies of Rips distance functions is bounded by the Gromov-Hausdorff distance of the underlying sets with tameness conditions and one of the set being bounded.

**Proposition D.7.** Let $X, Y \subset \mathbb{R}^d$ be two sets such that either $X$ or $Y$ is bounded, and $d_X^R$, $d_Y^R$ be the corresponding Rips distance functions. Suppose two sublevel filtrations $\{(d_X^R)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ and $\{(d_Y^R)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ are tame. Then the bottleneck distance between $\text{PH}_s(d_X^R)$ and $\text{PH}_s(d_Y^R)$ is upper bounded by the Gromov-Hausdorff distance as

$$d_B(\text{PH}_s(d_X^R), \text{PH}_s(d_Y^R)) \leq d_{GH}(X, Y).$$

### E Proofs for Section 2 and D

**Proof of Lemma 27** Note that $d_X^R(x) < r$ if and only if there exists $t$ with $s + t < r$, $\exists x_1, \ldots, x_n \in X$, $\exists y_i \in B_{d_X^R}(x_i, s)$ with $\|y_i - y_j\| < 2t$, $\forall i, j \leq n$, $x \in \text{conv}\{y_1, \ldots, y_n\}$. Hence $(d_X^R)^{-1}(-\infty, r)$ can be described as the announced statement.

**Proof of Proposition D.1** Fix $x, y \in X$, and let $r := d_X^R(x)$. Then $\exists s, t$ with $s + t < r$, $\exists x_1, \ldots, x_n \in X$, $\exists y_i \in B_{d_X^R}(x_i, s)$ with $\|x_i - x_j\| < 2t$, $\forall i, j \leq n$, $\exists \lambda_1, \ldots, \lambda_n \in [0, 1]$ with $\sum_{i=1}^n \lambda_i = 1$ and $x = \sum_{i=1}^n \lambda_i x_i$. For $i \leq n$, let $y_i := y - x + x_i$. Then $y_i \in B_{d_X^R}(x_i, s + \|x - y\|)$, and for all $i, j \leq n$, $\|y_i - y_j\| = \|x_i - x_j\| < 2t$. Also, $y$ is a convex combination of $y_i$'s as

$$y = (y - x) + (y - x) + \sum_{i=1}^n \lambda_i x'_i.$$
\[ d_R^B(x) = \sum_{i=1}^{n} \lambda_i (y - x_i) = \sum_{i=1}^{n} \lambda_i y_i. \]

Hence \( d_R^B(x) \leq r + \|x - y\| \). And by symmetry, the other inequality holds as well, and hence

\[ |d_R^B(x) - d_R^B(y)| \leq \|x - y\|. \]

\[ \Box \]

**Proof of Proposition [D.2]** Fix \( r > 0 \). Note that \( d_R^B(x) < r \) if and only if \( \exists s, t \) with \( s + t < r \), \( \exists x_1, \ldots, x_n \in X \), \( \exists y_i \in \mathbb{R}^{d}(x_i, s) \) with \( \|y_i - y_j\| < 2t \), \( \forall i, j \leq n \), \( x \in \text{conv}(\{y_1, \ldots, y_n\}) \). Hence once we define \( U_r(x) \) for each \( x \in X \) as

\[ U_r(x) := \left\{ z \in \mathbb{R}^d : \exists s, t \text{ with } s + t < r, \exists x_1, \ldots, x_n \in X \text{ with } x_1 = x, \exists y_i \in \mathbb{R}^{d}(x_i, s) \right. \]

with \( \|y_i - y_j\| < 2t \), \( \forall i, j \leq n \), \( z \in \text{conv}(\{y_1, \ldots, y_n\}) \),

then

\[ (d_R^B)^{-1}(-\infty, r) = \bigcup_{x \in X} U_r(x). \]

Also, \( U_r(x) \) is open in \( \mathbb{R}^d \): if \( z \in U_r(x) \), then there exists \( y_i \in \mathbb{R}^{d}(x_i, s) \), \( \lambda_i \in [0, 1] \) with \( \sum_{i=1}^{n} \lambda_i = 1 \) and \( z = \sum_{i=1}^{n} \lambda_i y_i \). Now, we can choose \( s' \) such that \( \mathbb{B}^{d}(y_i, s') \subset \mathbb{B}^{d}(x_i, s) \) for all \( i \). Then for all \( v \in \mathbb{B}^{d}(0, s') \),

\[ z + v = \sum_{i=1}^{n} \lambda_i (y_i + v), \]

and \( y_i + v \in \mathbb{B}^{d}(x_i, s) \), \( \|(y_i + v) - (y_j + v)\| < 2t \), and hence \( z + v \in U_r(x) \), i.e. \( \mathbb{B}^{d}(z, s') \subset U_r(x) \).

Also, note that \( \mathbb{B}^{d}(x, r) \subset U_r(x) \), so \( x \in U_r(x) \) and \( U_r(x) \) is nonempty. Now, consider the open cover \( \mathcal{U} \) of \( |R_X(r)| \) as

\[ \mathcal{U} := \{ U_r(x) : x \in X \}, \]

and consider its nerve \( \mathcal{N}\mathcal{U}_r \). Now, note that

\[ \cap_{i=1}^{k} U_r(x_i) := \left\{ z \in \mathbb{R}^d : \exists s, t \text{ with } s + t < r, \exists x_{k+1}, \ldots, x_n \in X, \exists y_i \in \mathbb{B}^{d}(x_i, s) \right. \]

with \( \|y_i - y_j\| < 2t \), \( \forall i, j \leq n \), \( z \in \text{conv}(\{y_1, \ldots, y_n\}) \).

Then \( \|x_i - x_j\| < 2r \), \( \forall i, j \leq k \) implies \( \text{conv}(\{x_1, \ldots, x_k\}) \subset \cap_{i=1}^{k} U_r(x_i) \) so \( \cap_{i=1}^{k} U_r(x_i) \neq \emptyset \). Also, \( \cap_{i=1}^{k} U_r(x_i) = \emptyset \) implies that there exists \( \exists s, t \) with \( s + t < r \), \( \exists x_{k+1}, \ldots, x_n \in X, \exists y_i \in \mathbb{B}^{d}(x_i, s) \) with \( \|y_i - y_j\| < 2t \), \( \forall i, j \leq n \). Then

\[ \|x_i - x_j\| \leq \|x_i - y_i\| + \|y_i - y_j\| + \|y_j - x_j\| < 2s + 2t < 2r, \]

hence in particular \( \|x_i - x_j\| < 2r, \forall i, j \leq k \) holds. Hence when \( \cap_{i=1}^{k} U_r(x_i) \) is mapped to \( \{x_1, \ldots, x_k\} \), the nerve complex \( \mathcal{N}\mathcal{U}_r \) equals the Rips complex \( R_X(r) \), i.e.

\[ \mathcal{N}\mathcal{U}_r = R_X(r). \quad (14) \]
Also, when \(|x_i - x_j| < 2r, \forall i, j \leq k\), define \(C_r(\{x_1, \ldots, x_k\}) := \text{conv}(\cup_{i=1}^k B_{2r}(x_i))\), then \(\cap_{i=1}^k U_r(x_i)\) deformation retracts to \(C_r(\{x_1, \ldots, x_k\})\), hence if \(\cap_{i=1}^k U_r(x_i)\) is nonempty then it is contractible. Also, when \(r \leq r'\), \(U_r(x) \subset U_{r'}(x)\) as well. Hence from [13] Lemma 3.4, there exists \(\phi_r : \bigcup_x U_r(x) \to NU_r\) and \(\psi_r : NU_r \to \bigcup_x U_r(x)\) such that for \(r \leq r'\), the following diagram commutes on the homotopy or homology level with the inclusion map:

\[
\begin{array}{ccc}
\bigcup_x U_r(x) & \longrightarrow & \bigcup_x U_{r'}(x) \\
\phi_r & \Downarrow & \psi_r \\
NU_r & \longrightarrow & NU_{r'}
\end{array}
\]

Then from Lemma 21, \(\phi_r\) and \(\psi_r\) can be understood as maps between \((d^R_\mathcal{X})^{-1}(-\infty, r)\) and \(R^\mathcal{X}(r)\) as well, i.e., \(\phi_r : (d^R_\mathcal{X})^{-1}(-\infty, r) \to R^\mathcal{X}(r)\) and \(\psi_r : R^\mathcal{X}(r) \to (d^R_\mathcal{X})^{-1}(-\infty, r)\) as well.

**Proof of Proposition D.3.** The map \(\phi_r\) and \(\psi_r\) in Proposition D.2 gives the homotopy between \((d^R_\mathcal{X})^{-1}(-\infty, r)\) and \(R^\mathcal{X}(r)\).

**Proof of Proposition D.4.** Also, [13] gives that \(\{ (d^R_\mathcal{X})^{-1}(-\infty, r) \}_{r \in \mathbb{R}}\) and \(\{ R^\mathcal{X}(r) \}_{r \in \mathbb{R}}\) are \(\epsilon\)-interleaved for any \(\epsilon > 0\), and hence \(d_H(d^R_\mathcal{X}, R^\mathcal{X}) = 0\), which implies that \(PH_*(d^R_\mathcal{X}) = PH_*(R^\mathcal{X})\).

**Proof of Proposition D.5.** Let \(A \subset \mathbb{R}\) be a compact, then there exists \(r := \max A\). From Proposition D.1 \(d^R_\mathcal{X}\) is continuous, and hence \((d^R_\mathcal{X})^{-1}(-\infty, r)\) is closed. And from Lemma 21

\[(d^R_\mathcal{X})^{-1}(-\infty, r] \subset (d^R_\mathcal{X})^{-1}(-\infty, r + 1) \subset \{ x \in \mathbb{R}^d : d(x, \text{conv}(\mathcal{X})) < r + 1 \}.

Hence if \(\mathcal{X}\) is bounded, then \((d^R_\mathcal{X})^{-1}(-\infty, r]\) is also a bounded set. Hence \((d^R_\mathcal{X})^{-1}(-\infty, r]\) is compact. Then \((d^R_\mathcal{X})^{-1}(A)\) is a closed set inside a compact set, and hence is compact as well, i.e. \(d^R_\mathcal{X}\) is proper.
Now, $d^R_{\mathcal{X}} \geq 0$, so $d^R_{\mathcal{X}}$ is bounded below. Hence with $d^R_{\mathcal{X}}$ being proper, from [17, Corollary 3.34], the sublevel filtration $\{(d^R_{\mathcal{X}})^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ is tame.

The proof of Proposition [D.6] is a direct consequence of Proposition [D.4].

**Proof of Proposition [D.6]** From Proposition [D.4], two sublevel filtrations $\{(d^R_{\mathcal{X}})^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ and $\{(d^R_{\mathcal{Y}})^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ are interleaved by $d_H(\mathcal{X}, \mathcal{Y})$. And since two filtrations are tame, from Stability Theorem for persistence modules ([11, Theorem 4.4]), $d_B(\text{PH}_*(d^R_{\mathcal{X}}), \text{PH}_*(d^R_{\mathcal{Y}})) \leq d_H(\mathcal{X}, \mathcal{Y})$ holds.

Proposition [D.7] is by combining the stability theorem for Rips complex filtration in [11, Lemma 4.3] and Proposition [D.6].

**Proof of Proposition [D.7]** We can assume without loss of generality that $\mathcal{X}$ is bounded. If $\mathcal{Y}$ is not bounded, then $d_{GH}(\mathcal{X}, \mathcal{Y}) = \infty$ and there is nothing to prove. Hence we can assume that $\mathcal{Y}$ is also bounded.

Fix any $\epsilon > 0$, and let $\mathcal{X}_\epsilon \subset \mathcal{X}$ and $\mathcal{Y}_\epsilon \subset \mathcal{Y}$ be finite $\epsilon$-covers of $\mathcal{X}$ and $\mathcal{Y}$, respectively, i.e. satisfying $d_H(\mathcal{X}_\epsilon, \mathcal{X}) < \epsilon$ and $d_H(\mathcal{Y}_\epsilon, \mathcal{Y}) < \epsilon$. Then $d_B(\text{PH}_*(d^R_{\mathcal{X}}), \text{PH}_*(d^R_{\mathcal{Y}}))$ can be factorized as

$$d_B(\text{PH}_*(d^R_{\mathcal{X}}), \text{PH}_*(d^R_{\mathcal{Y}}))$$

$$\leq d_B(\text{PH}_*(d^R_{\mathcal{X}_\epsilon}), \text{PH}_*(d^R_{\mathcal{Y}_\epsilon})) + d_B(\text{PH}_*(d^R_{\mathcal{X}_\epsilon}), \text{PH}_*(d^R_{\mathcal{Y}_\epsilon})) + d_B(\text{PH}_*(d^R_{\mathcal{X}_\epsilon}), \text{PH}_*(d^R_{\mathcal{Y}_\epsilon})).$$

(16)

For the first term of (16), since $\mathcal{X}_\epsilon, \mathcal{Y}_\epsilon$ are both finite, $\text{PH}_*(d^R_{\mathcal{X}_\epsilon}) = \text{PH}_*(R_{\mathcal{X}_\epsilon})$ and $\text{PH}_*(d^R_{\mathcal{Y}_\epsilon}) = \text{PH}_*(R_{\mathcal{Y}_\epsilon})$ from Theorem [8]. And hence from Rips complex filtration in [11, Lemma 4.3],

$$d_B(\text{PH}_*(d^R_{\mathcal{X}_\epsilon}), \text{PH}_*(d^R_{\mathcal{Y}_\epsilon})) = d_B(\text{PH}_*(R_{\mathcal{X}_\epsilon}), \text{PH}_*(R_{\mathcal{Y}_\epsilon}))$$

$$\leq d_{GH}(\mathcal{X}_\epsilon, \mathcal{Y}_\epsilon).$$

Note that constant is different due to our Rips complex definition being different from [11]. Then since Gromov-Hausdorff distance is bounded by Hausdorff distance,

$$d_{GH}(\mathcal{X}_\epsilon, \mathcal{Y}_\epsilon) \leq d_{GH}(\mathcal{X}, \mathcal{Y}) + d_{GH}(\mathcal{X}_\epsilon, \mathcal{X}) + d_{GH}(\mathcal{Y}_\epsilon, \mathcal{Y})$$

$$\leq d_{GH}(\mathcal{X}, \mathcal{Y}) + d_H(\mathcal{X}_\epsilon, \mathcal{X}) + d_H(\mathcal{Y}_\epsilon, \mathcal{Y})$$

$$< d_{GH}(\mathcal{X}, \mathcal{Y}) + 2\epsilon,$$

which gives

$$d_B(\text{PH}_*(d^R_{\mathcal{X}_\epsilon}), \text{PH}_*(d^R_{\mathcal{Y}_\epsilon})) < d_{GH}(\mathcal{X}, \mathcal{Y}) + 2\epsilon.$$

For the second and third terms of (16), since $\mathcal{X}_\epsilon$ and $\mathcal{Y}_\epsilon$ are finite, from Proposition [D.5], their sublevel filtrations $\{(d^R_{\mathcal{X}_\epsilon})^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ and $\{(d^R_{\mathcal{Y}_\epsilon})^{-1}(-\infty, r)\}_{r \in \mathbb{R}}$ are both tame. Hence with the tameness conditions of $d^R_{\mathcal{X}}$ and $d^R_{\mathcal{Y}}$, by Proposition [D.6],

$$d_B(\text{PH}_*(d^R_{\mathcal{X}_\epsilon}), \text{PH}_*(d^R_{\mathcal{X}_\epsilon})) \leq d_H(\mathcal{X}_\epsilon, \mathcal{X}) < \epsilon,$$

$$d_B(\text{PH}_*(d^R_{\mathcal{Y}_\epsilon}), \text{PH}_*(d^R_{\mathcal{Y}_\epsilon})) \leq d_H(\mathcal{Y}_\epsilon, \mathcal{Y}) < \epsilon.$$
And hence (16) is bounded as
\[
\begin{align*}
&d_B(PH_*(d^R_X), PH_*(d^R_Y)) \\
&\leq d_B(PH_*(d^R_X), PH_*(d^R_Y)) + d_B(PH_*(d^R_X), PH_*(d^R_Y)) + d_B(PH_*(d^R_X), PH_*(d^R_Y)) \\
&< d_{GH}(X, Y) + 2\epsilon + \epsilon + \epsilon \\
&= d_{GH}(X, Y) + 4\epsilon.
\end{align*}
\]

Since the choice of \(\epsilon > 0\) was arbitrary, we have the desired bound as
\[
d_B(PH_*(d^R_X), PH_*(d^R_Y)) \leq d_{GH}(X, Y).
\]

Then Theorem \[8\] is a Corollary from Proposition \[D.7\] and Proposition \[D.5\].

**Proof of Theorem 8.** Since \(X\) and \(Y\) are both bounded, from Proposition \[D.5\], their sublevel filtrations \(\{(d^R_X)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}\) and \(\{(d^R_Y)^{-1}(-\infty, r)\}_{r \in \mathbb{R}}\) are both tame. Hence applying Proposition \[D.7\] concludes the proof.

**Proof of Corollary 2.1.** Since \(X\) and \(Y\) are both bounded, applying Theorem 8 gives that
\[
d_B(PH_*(d^R_X), PH_*(d^R_Y)) \leq d_{GH}(X, X).
\]

Then since \(X\) is finite, applying Proposition 2.1 gives \(PH_*(d^R_X) = PH_*(R_X)\), and hence
\[
d_B(PH_*(R_X), PH_*(d^R_X)) = d_B(PH_*(d^R_X), PH_*(d^R_X)) \leq d_{GH}(X, X),
\]
which shows the statement.

**Proof of Theorem 9.** Let \(k, n \in \mathbb{N}\) be given, and use notation \(X_n = \{X_1, \ldots, X_n\}\). Since \(\text{supp}(P)\) is bounded, there exists \(N = N_k < \infty\) and \(x_1, \ldots, x_N \in \text{supp}(P)\) satisfying
\[
\text{supp}(P) \subset \bigcup_{j=1}^{N} B_{\mathbb{R}^d}(x_j, \frac{1}{2k}).
\]

Let \(E_{k,n}\) be the event that all \(B_{\mathbb{R}^d}(x_j, \frac{k}{2})\) have intersections with \(\{X_1, \ldots, X_n\}\), i.e. for each \(1 \leq j \leq N\), there exists \(1 \leq i \leq n\) with \(x_j \in B_{\mathbb{R}^d}(x_i, \frac{1}{2k})\). From \(B_{\mathbb{R}^d}(x_j, \frac{1}{2k}) \subset B_{\mathbb{R}^d}(X_i, \frac{1}{k})\), under the event \(E_{k,n}\),
\[
\text{supp}(P) \subset \bigcup_{j=1}^{N} B_{\mathbb{R}^d}(x_j, \frac{1}{2k}) \subset \bigcup_{i=1}^{n} B_{\mathbb{R}^d}(X_i, \frac{1}{k}).
\]

Also, \(\{X_1, \ldots, X_n\} \subset \text{supp}(P)\) a.s., and hence under the event \(E_{k,n}\),
\[
d_{GH}(X_n, \text{supp}(P)) \leq \frac{1}{k}.
\]
Hence under the event $E_{k,n}$, since both $X_n$ and $\text{supp}(P)$ are bounded, from Theorem 8

$$d_B(PH_*(R_{X_n}), PH_*(d_{\text{supp}(P)}^R)) \leq d_{GH}(X_n, \text{supp}(P)) \leq \frac{1}{k}. \tag{17}$$

Hence the probability of the complementary of the event $E_{k,n}$ is upper bounded as

$$P\left(\overline{E_{k,n}}\right) = P\left(\left(\bigcap_{j=1}^{N} \bigcup_{i=1}^{n} \left\{ X_i \not\in B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right\} \right)^c\right)$$

$$= P\left(\bigcup_{j=1}^{N} \bigcap_{i=1}^{n} \left\{ X_i \not\in B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right\}\right)$$

$$\leq \sum_{j=1}^{N} P\left(\bigcap_{i=1}^{n} \left\{ X_i \not\in B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right\}\right)$$

$$= \sum_{j=1}^{N} \prod_{i=1}^{n} P\left( X_i \not\in B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right)$$

$$= \sum_{j=1}^{N} \left( 1 - P\left( B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right) \right)^n$$

$$\leq \sum_{j=1}^{N} \exp\left( -nP\left( B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right) \right). \tag{18}$$

Now, note that since $x_j \in \text{supp}(P)$, $P\left( B_{\mathbb{R}^d} \left( x_j, \frac{1}{2k} \right) \right) > 0$ for all $1 \leq j \leq N$. Hence for each $k \in \mathbb{N}$, there exists $n_k \in \mathbb{N}$ such that $P\left( E_{k,n_k}^c \right) \leq \frac{1}{k^2}$. Then

$$\sum_{k=1}^{\infty} P\left( E_{k,n_k}^c \right) \leq \sum_{k=1}^{\infty} \frac{1}{k^2} < \infty.$$

Hence from Borel-Cantelli lemma, the events $\left\{ E_{k,n_k}^c \right\}_{k \in \mathbb{N}}$ happening infinitely often is probability 0, i.e. $P\left( \bigcap_{K=1}^{\infty} \bigcup_{k=K}^{\infty} E_{k,n_k}^c \right) = 0$. Hence the events $\left\{ E_{k,n_k} \right\}_{k \in \mathbb{N}}$ happen all but finite, i.e.

$$P\left( \bigcup_{K=1}^{\infty} \bigcup_{k=K}^{\infty} E_{k,n_k} \right) = 1. \tag{19}$$

Now, for $\omega \in \bigcup_{K=1}^{\infty} \bigcap_{k=K}^{\infty} E_{k,n_k}$, there exists $K \in \mathbb{N}$ such that $\omega \in \bigcap_{k=K}^{\infty} E_{k,n_k}$. Fix $\epsilon > 0$ and choose $k \geq K$ with $\frac{1}{k} < \epsilon$. Then from (17) and that $E_{k,n_k} \subset E_{k,n}$ for $n \geq n_k$,

$$d_B(PH_*(R_{X_n}(\omega)), PH_*(d_{\text{supp}(P)}^R)) \leq \frac{1}{k} < \epsilon$$

and hence $d_B(PH_*(R_{X_n}(\omega)), PH_*(d_{\text{supp}(P)}^R)) \to 0$ for any fixed $\omega \in \bigcup_{K=1}^{\infty} \bigcap_{k=K}^{\infty} E_{k,n_k}$. Hence from (19),

$$d_B(PH_*(R_{X_n}), PH_*(d_{\text{supp}(P)}^R)) \to 0 \text{ a.s.}$$
F Proofs of Section 4

Proof of Proposition 4.1. When \( \|\epsilon\|_\infty = \infty \), then there is nothing to prove, so we can assume that \( \|\epsilon\|_\infty < \infty \). Then since \( f \) is a Lipschitz function on a bounded domain \([0, T]\), \( X \) is bounded as well.

Let \( X^N := \{SW_{m,\tau}f(t) : t = \delta_T(1 + (m - 1)\tau), \delta_T(2 + (m - 1)\tau), \ldots, T\} \) and write as a matrix where each element corresponds to each row. Denote \( \tilde{i} \)th rows of \( X \) and \( X^N \) as \( X_i \) and \( X_i^N \). Let \( \tilde{N} := N - (m - 1)\tau \), and understand \( X \) and \( X^N \) as a point cloud of \( \{X_i\}_{i \leq \tilde{N}} \) and \( \{X_i^N\}_{i \leq \tilde{N}} \), respectively, depending on the context.

We first consider bounding the Gromov-Hausdorff distance \( d_{GH}(X^l, X) \) between the data after PCA \( X^l \) and the Takens embedding of the signal \( X \). Note that applying PCA is isometric to \( X \) being projected to linear subspaces generated by first \( l \) eigenvectors of \( \frac{1}{N}X^\top X \). For \( j \leq d \), let \( V^k \) and \( V^k \) be the linear subspaces generated by first \( k \) eigenvectors of \( \frac{1}{N}(X^N)^\top X^N \) and \( \frac{1}{N}X^\top X \), respectively, and let \( \Pi^k \) and \( \Pi^k \) be the projection operator to \( V^k \) and \( V^k \), respectively. With this notation, \( X^l \) is isometric to \( \Pi^l(X) \), where \( X \) is understood as a point cloud. And hence \( d_{GH}(X^l, X) \) can be bounded as

\[
d_{GH}(X^l, X) = d_{GH}(\Pi^l(X), X) \\
\leq d_H(\Pi^l(X), X) \\
\leq d_H(\Pi^l(X), X^N) + d_H(X^N, X). \tag{20}
\]

Now, consider the first term of \( \tag{20} \), which is coming from the error and PCA. First, from \cite[Theorem 2]{64},

\[
\|(Id - \Pi^l)\Pi^l\|_F \leq \frac{2}{\lambda_l} \left\| \frac{1}{N}X^\top X - \frac{1}{N}(X^N)^\top X^N \right\|_F, \\
\leq \frac{2 \sum_{i=1}^{\tilde{N}} \|X_i^\top X_i - (X_i^N)^\top X_i^N\|_F}{N\lambda_l}. \tag{21}
\]

Now, let \( \epsilon_i := (SW_{m,\tau}\epsilon(i), SW_{m,\tau}\epsilon(i+\tau), \ldots, SW_{m,\tau}\epsilon(i+(m - 1)\tau)) \) and understand as a row vector, so that \( X_i = X_i^N + \epsilon_i \). Then

\[
\|X_i^\top X_i - (X_i^N)^\top X_i^N\|_F = \left\| (X_i^N + \epsilon_i)^\top (X_i^N + \epsilon_i) - (X_i^N)^\top X_i^N \right\|_F \\
= \left\| (X_i^N)^\top \epsilon_i \right\|_F + \left\| \epsilon_i^\top X_i^N \right\|_F + \left\| \epsilon_i^\top \epsilon_i \right\|_F \\
\leq 2 \|X_i^N\|_F \|\epsilon_i\|_2 + \|\epsilon_i\|_2^2.
\]

Then since each element of \( X_i^N \) is bounded by \( LfT \) and each element of \( \epsilon_i \) is bounded by \( \|\epsilon\|_\infty \), applying to \( \tag{21} \) gives the bound as

\[
\|(Id - \Pi^l)\Pi^l\|_F \leq \frac{m \|\epsilon\|_\infty (4LfT + \|\epsilon\|_\infty)}{\lambda_l}.
\]

Now, note that the distance between \( \Pi^l(X_i) \) (after PCA) and \( X_i^N \) can be bounded as

\[
\|\Pi^l(X_i) - X_i^N\|_2 = \|\Pi^l(\epsilon_i) + \Pi^l(X_i^N) - X_i^N\|_2
\]

34
can be bounded as
Since $X \in V^l$ implies that $\|(Id - \Pi_{V^l})v\|_2 \leq \|(Id - \Pi_{V^l})v\|_2$, and $X^N_i \in V^l$ implies $\Pi_{V^l}(X^N_i) = X^N_i$, and hence the first term of (22) is bounded as

$$\|(Id - \Pi_{V^l})(X^N_i)\|_2 \leq \|(Id - \Pi_{V^l})(X^N_i)\|_2 = \|(Id - \Pi_{V^l})\Pi_{V^l}(X^N_i)\|_2 \leq \|(Id - \Pi_{V^l})\Pi_{V^l}\|_2 \|X^N_i\|_2.$$ 

Then further applying (21) and $\|X^N_i\|_2 \leq \sqrt{m}L_fT$ gives

$$\|(Id - \Pi_{V^l})(X^N_i)\|_2 \leq \frac{m^3}{\lambda_f}L_fT\|\epsilon\|_{\infty}(4L_fT + \|\epsilon\|_{\infty}).$$ 

Now, the second term of (22) is bounded as

$$\|\Pi_{V^l}(\epsilon)\|_2 \leq \|\epsilon\|_2 \leq \sqrt{m} \|\epsilon\|_{\infty}.$$ 

Then by applying (23) and (24) to (22), the Hausdorff distance between $\Pi_{V^l}(X)$ and $X^N$ can be bounded as

$$d_H(\Pi_{V^l}(X), X^N) \leq \sup_{1 \leq i \leq N} \|\Pi_{V^l}(X_i) - X^N_i\|_2 \leq \sqrt{m} \|\epsilon\|_{\infty} + \frac{m^3}{\lambda_f}L_fT\|\epsilon\|_{\infty}(4L_fT + \|\epsilon\|_{\infty}).$$ 

Now, consider the second term of (20), which is coming from the time sampling. Since we have assumed that $f$ is $L_f$-Lipschitz, i.e. $|f(t_1) - f(t_2)\| \leq L_f|t_1 - t_2|$, $SW_{m,\tau}f$ is also Lipschitz with constant $\sqrt{m}L_f$ as

$$\|SW_{m,\tau}f(t_1) - SW_{m,\tau}f(t_2)\|_2 = \sqrt{\sum_{k=0}^{m-1} \|f(t_1 + k(m-1)\tau) - f(t_2 + k(m-1)\tau)\|^2} \leq \sqrt{m}L_f|t_1 - t_2|.$$ 

Since $X^N$ is sampled from $X$ by sampling grid size $\delta_T = \frac{T}{N-1}$, $d_H(X^N, X)$ is bounded as

$$d_H(X^N, X) \leq \frac{\sqrt{m}L_fT}{N}.$$ 

Hence by applying (25) and (26) to (20), the Gromov-Hausdorff distance $d_{GH}(X^l, X)$ can be bounded as

$$d_{GH}(X^l, X) \leq d_H(\Pi_{V^l}(X), X^N) + d_H(X^N, X) \leq \sqrt{m} \|\epsilon\|_{\infty} + \frac{m^3}{\lambda_f}L_fT\|\epsilon\|_{\infty}(4L_fT + \|\epsilon\|_{\infty}) + \sqrt{m}L_fT$$ 

Now, from the assumption $\|\epsilon\|_{\infty} < \infty$, both $X^l$ and $X$ are bounded, hence applying (27) to Theorem 8 bounds the bottleneck distance $d_B(D_X, D_f)$ as

$$d_B(D_X, D_f) \leq d_{GH}(X^l, X)$$ 

35
\[
\leq \sqrt{m} \|\epsilon\|_\infty + \frac{m^{\frac{3}{2}} L_f T \|\epsilon\|_\infty (4L_f T + \|\epsilon\|_\infty)}{\lambda_l} + \frac{\sqrt{mL_f T}}{N}.
\]

\[\Box \]

Proof of Theorem 14. From Theorem A.1 in [5] and Proposition 4.1,
\[
d_B(\lambda_{k,X}, \lambda_{k,f}) \leq d_B(D_X, D_f) \leq \sqrt{m} \|\epsilon\|_\infty + \frac{m^{\frac{3}{2}} L_f T \|\epsilon\|_\infty (4L_f T + \|\epsilon\|_\infty)}{\lambda_l} + \frac{\sqrt{mL_f T}}{N}.
\]

And since silhouette is weighted sum of \(\Lambda_p\), it is also correspondingly bounded as
\[
d_B(\phi^{(p)}_X, \phi^{(p)}_f) \leq d_B(D_X, D_f) \leq \sqrt{m} \|\epsilon\|_\infty + \frac{m^{\frac{3}{2}} L_f T \|\epsilon\|_\infty (4L_f T + \|\epsilon\|_\infty)}{\lambda_l} + \frac{\sqrt{mL_f T}}{N}.
\]

\[\Box \]