Topological Classification of Time Oscillations

Gabriel Riegner

12 June 2024

0.1 Introduction

The detection of periodicity in timeseries has broad applications. An example is detecting brain oscillations: rhythmic patterns that arise from synchronized activity across large ensembles of neurons, and measurable using electroencephalography. In recordings of electrical activity, neural oscillations often occur spontaneously. Detecting when oscillations occur is an important part of neural signal analysis. Traditional methods often rely on analyzing signals in the frequency domain, using Fourier series to decompose the timeseries into sums of sine waves [DSV]. However, neural signals are not necessarily sinusoidal, so traditional methods may not fully capture their complex topological features. This project proposes a novel approach for classifying timeseries as periodic (oscillatory) or aperiodic (non-oscillatory) using topological data analysis (TDA) to extract features for classification.

The key idea is to leverage the topological properties of timeseries embeddings [SBDH, Tak] and autocorrelation functions to distinguish between periodic and aperiodic signals. A concrete starting point are simple linear time processes, specifically autoregressive order two (ar-2) processes. In an ar-2 process, each timepoint is a linear combination of its two previous timepoints, and the underlying parameters determine whether it exhibits oscillatory behavior. As a training dataset for classification, I simulate ar-2 processes, because the choice of theoretical parameters leads to well-defined oscillatory dynamics. Distinct topological features are then computed from the methods of persistent homology, including Vietoris-Rips filtrations and Sublevel-Set filtrations. Finally, features from the resulting persistence diagrams–amplitudes, persistence entropies, number of points–are extracted for use in a logistic regression classification of periodic versus aperiodic signals. The accuracies of the fitted models are compared against the theoretical boundary between periodic and aperiodic stationary ar-2 processes.

0.1.1 TDA Pipeline

The proposed classification pipeline consists of the following steps:

1. **Generate periodic and aperiodic timeseries** and their corresponding autocorrelation functions.
2. **Create persistence diagrams** from both the timeseries [ELZ] and autocorrelation functions.

3. **Convert these persistence diagrams into vectors** using measures like amplitude, persistence entropy, and number of points [TLT$^+$, AGDR].

4. **Train a Logistic Regression classifier** using the extracted topological features from both time and autocorrelation domains.

5. **Evaluate and compare classification accuracies** of these methods in distinguishing periodic from non-periodic signals.

0.2 Experiments

0.2.1 Autoregressive Timeseries

To generate the time series data used in this study we used an ar-2 process:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + e_t, \quad e_t \overset{\text{iid}}{\sim} \mathcal{N}(0,1). \tag{1}$$

Here, x_t denotes the value of the time series at time t, ϕ_1 and ϕ_2 are the coefficients that determine the influence of the past two timepoints on the current timepoint, and e_t is an independent and identically distributed (iid) Gaussian noise term.

The behavior of the ar-2 process is determined by the values of the coefficients ϕ_1 and ϕ_2. Specifically, whether the process is stationary and whether it exhibits periodic behavior depends on these coefficients. The stationarity region in the (ϕ_1, ϕ_2) plane is defined by the conditions $\phi_2 < 1 + \phi_1$, $\phi_2 < 1 - \phi_1$, and $\phi_2 > -1$. Within this region, the theoretical boundary between periodic and aperiodic behavior is given by $\phi_2 = -\phi_1^2/4$.

To simulate the data, we generate a grid of (ϕ_1, ϕ_2) pairs within the stationarity region and classify each pair as corresponding to either a periodic or aperiodic process based on the theoretical boundary. This provides us with a labeled dataset of timeseries for subsequent analysis.

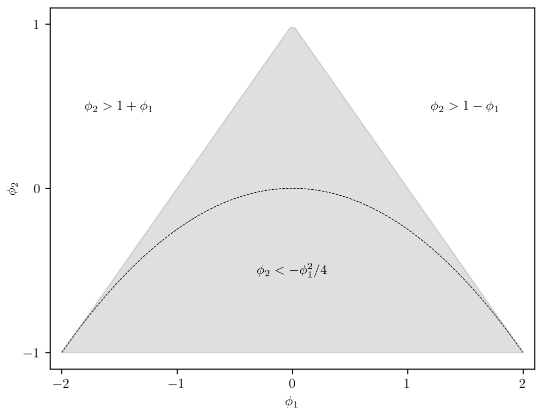


Figure 1: Stationarity triangle for AR(2) processes.

0.2.2 Time Domain

In the time domain analysis, we used Taken's embedding theorem [SBDH] to transform each one-dimensional timeseries into a higher-dimensional space. Taken's embedding maps the timeseries x_t into a set of vectors in $\mathbb{R}^d$, where d is the embedding dimension and τ is the time delay. Formally, the embedding is defined as:

$$\mathrm{emb}(x_t) \mapsto \{x_t, x_{t-\tau}, x_{t-2\tau}, \ldots, x_{t-(d-1)\tau}\}, \tag{2}$$

This embedding helps in reconstructing the phase space of the timeseries, making it easier to analyze its topological features.

Once the timeseries is embedded, we apply the Vietoris-Rips filtration to the resulting point cloud in $\mathbb{R}^d$. A Vietoris-Rips complex is a simplicial complex constructed by connecting points that are within a certain distance s from each other. A subset of the embedded points forms a simplex if all pairwise distances are $\leq s$:

$$\mathrm{VR}_s(\mathrm{emb}(x_t)) = \{[v_0, \ldots, v_n] \mid \forall i, j; d(v_i, v_j) \leq s\} \tag{3}$$

The Vietoris-Rips filtration consists of a sequence of these complexes for increasing values of s, capturing the multi-scale topological features of the data. However, the figure 2 shows that topological features like connected components and loops or cycles are difficult to capture do to noise in the autoregressive process.

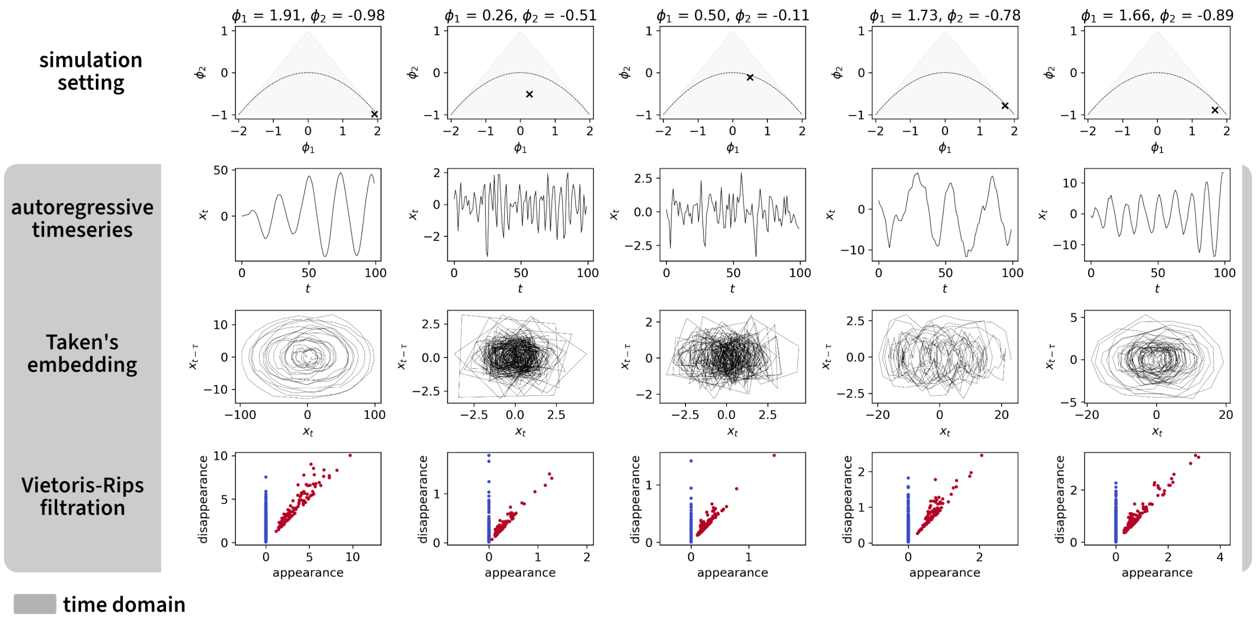


Figure 2: Time Domain Topology of Periodic AR(2) processes.

0.2.3 Autocorrelation Domain

In addition to the time domain analysis, we also analyze the autocorrelation domain which helps to reduce noise and provide a clearer indication of periodic patterns (figure 3). The autocorrelation function ρ_k measures the correlation between the timeseries values at different lags k. For a stationary time series, the autocorrelation function depends only on the lag:

$$\rho_k = \mathrm{corr}(x_t, x_{t-k}), \tag{4}$$

To capture the topological features of the autocorrelation function, we apply sublevel set filtration, which provides a stable description of the critical points of the autocorrelation function. The sublevel set of a discrete function at a threshold a is the set of points where the function values are $\leq a$:

$$\mathrm{SL}a(\rho) = \{k \mid \rho_k \leq a\}. \tag{5}$$

As the threshold a increases, we obtain a sequence of nested sublevel sets, forming a filtration:

$$\mathrm{SL}_{a_1}(\rho) \subseteq \mathrm{SL}_{a_2}(\rho) \subseteq \ldots \subseteq \mathrm{SL}_{a_n}(\rho) \quad \text{for} \quad a_1 \leq a_2 \leq \ldots \leq a_n \tag{6}$$

We also apply Taken's embedding to the autocorrelation function to further analyze its topological features. This involves embedding the autocorrelation function values into a higher-dimensional space:

$$\mathrm{emb}(\rho_k) \mapsto \{\rho_k, \rho_{k-\tau}, \rho_{k-2\tau}, \ldots, \rho_{k-(d-1)\tau}\}, \tag{7}$$

Similar to the time domain analysis, we then apply the Vietoris-Rips filtration to the embedded points of the autocorrelation function:

$$\mathrm{VR}_s(\mathrm{emb}(\rho_k)) = \{[v_0, \ldots, v_n] \mid \forall i, j; d(v_i, v_j) \leq s\}. \tag{8}$$

A property of stationary ar-2 processes is that their autocorrelations decay to zero, and as the amplitude of oscillations decreases, the embeddings spirals to the origin 3.

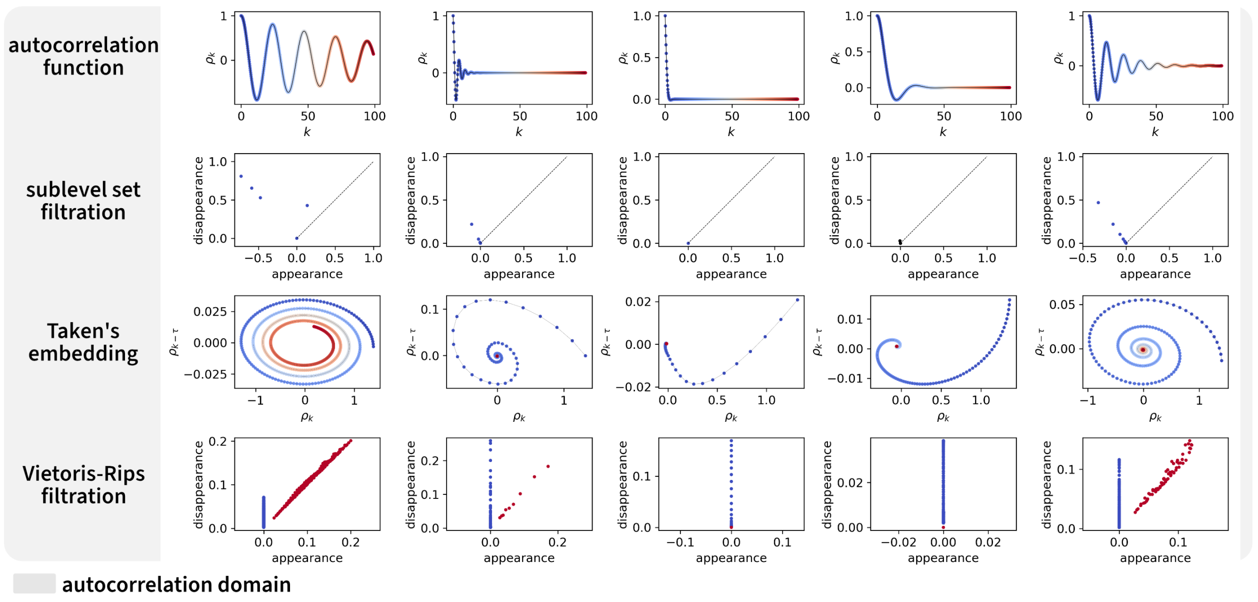


Figure 3: Autocorrelation Domain Topology of Periodic AR(2) processes.

0.2.4 Vectorization of Persistence Diagrams

After generating the persistence diagrams from both the time and autocorrelation domains, we convert these diagrams into feature vectors that can be used in a classifier. In this project, we use three methods for vectorization, as implemented in the *giotto-tda* library [TLT$^+$]:

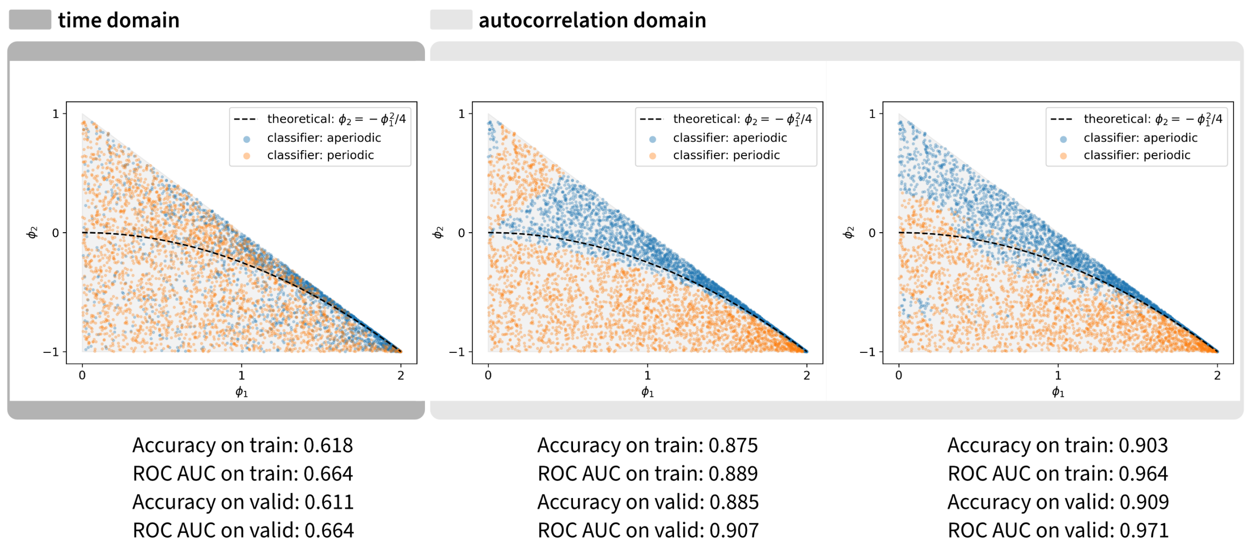
1. **Amplitude**: The amplitude is calculated as the L^2 distance between persistence landscapes, which are piecewise-linear functions representing the persistence diagram. This measure captures the overall magnitude of the topological features.

2. **Persistence entropy**: Persistence entropy is calculated as the Shannon entropy of the points on a persistence diagram. This measure captures the complexity or uncertainty associated with the distribution of the topological features.

3. **Number of points**: This measure simply counts the number of off-diagonal points in the persistence diagrams, providing a straightforward summary of the topological features present in the data.

0.3 Results

We evaluated the performance of a logistic regression classifier trained on the topological feature vectors derived from both the time and autocorrelation domains by comparing its predictions against the theoretical classification boundary of the ar-2 processes. The simulation settings included $T = 250$ timepoints and $K = 50$ autocorrelation lags. The classifier was an L^2 logistic regression, trained on a dataset of $N = 4000$ samples and validated on a set of $N = 1000$ samples. Performance metrics included accuracy, defined as the fraction of correct classifications, and the area under the ROC curve, which evaluates the trade-off between true positives and false positives.



Accuracy on train: 0.618
ROC AUC on train: 0.664
Accuracy on valid: 0.611
ROC AUC on valid: 0.664

Accuracy on train: 0.875
ROC AUC on train: 0.889
Accuracy on valid: 0.885
ROC AUC on valid: 0.907

Accuracy on train: 0.903
ROC AUC on train: 0.964
Accuracy on valid: 0.909
ROC AUC on valid: 0.971

Figure 4: Classification performance results.

Key results from figure 4 include:

1. **Time domain, Taken's embedding, Vietoris-Rips filtration** performed poorly, likely attributed to the presence of noise embeddings. This suggests that more robust methods for estimating topological features from timeseries are needed, such as weighted Rips persistence [TLT+].

2. **Autocorrelation domain, Sublevel Set filtration** showed a large accuracy improvement in accuracy compared to the time domain approach. This underscores the importance of in making periodicity more detectable.

3. **Autocorrelation domain, Taken's embedding, Vietoris-Rips filtration** was the most effective method. This suggests that mapping the timeseries through multiple stages (from time to autocorrelation, then embedding) captures more robust topological features that are effective for classifying periodicity.

In summary, this project presents approaches to classifying timeseries as periodic or aperiodic using topological data analysis (TDA). Persistence diagrams resulting from Vietoris-Rips and sublevel set filtrations to both time and autocorrelation domains can summarize information about the appearance and disappearance of topological features (homology classes) at various dimensions, including connected components and cycles. This approach gives insights into the geometric and topological features that may not be evident in the frequency domain. Additionally, we demonstrate the impact of incorporating different topological features on the accuracy of classifying time oscillations, underscoring the potential of TDA in enhancing our understanding of complex temporal dynamics in neuroscience and beyond.

References

[AGDR] Nieves Atienza, Rocio Gonzalez-Diaz, and Matteo Rucco. Persistent entropy for separating topological features from noise in vietoris-rips complexes. 52(3):637–655.

[DSV] Thomas Donoghue, Natalie Schaworonkow, and Bradley Voytek. Methodological considerations for studying neural oscillations. 55(11):3502–3527.

[ELZ] Edelsbrunner, Letscher, and Zomorodian. Topological persistence and simplification. 28(4):511–533.

[SBDH] J. Stark, D.S. Broomhead, M.E. Davies, and J. Huke. Takens embedding theorems for forced and stochastic systems. 30(8):5303–5314.

[Tak] Floris Takens. Detecting strange attractors in turbulence. In David Rand and Lai-Sang Young, editors, *Dynamical Systems and Turbulence, Warwick 1980*, volume 898, pages 366–381. Springer Berlin Heidelberg. Series Title: Lecture Notes in Mathematics.

[TLT$^+$] Guillaume Tauzin, Umberto Lupo, Lewis Tunstall, Julian Burella Pérez, Matteo Caorsi, Wojciech Reise, Anibal Medina-Mardones, Alberto Dassatti, and Kathryn Hess. giotto-tda: A topological data analysis toolkit for machine learning and data exploration.

Project Code

https://github.com/griegner/dsc214