(supplementary material) tramoTDA: A Trajectory Monitoring System Using Topological Data Analysis

Miriam Esteve[a], Antonio Falcó[b]

[a]*Department of Matematicas, Fisica y Ciencias Tecnologicas, Universidad Cardenal Herrera-CEU, CEU Universities, Spain, miriam.estevecampello@uchceu.es*
[b]*Department of Matematicas, Fisica y Ciencias Tecnologicas, Universidad Cardenal Herrera-CEU, CEU Universities,Spain, afalco@uchceu.es*

1. Methodology

1.1. Simulated Trajectory Data Generation

tramoTDA includes a robust module for generating simulated trajectory data. This is particularly useful for testing and validating TDA processes under controlled conditions. The module can simulate a variety of trajectory patterns, ensuring comprehensive evaluation of the software's analytical capabilities. The generated data serves as a consistent and reliable basis for demonstrating the functionality and effectiveness of *tramoTDA* in various application scenarios. The module generates four types of trajectory patterns: Brownian motion, Lévy flight, spiral trajectories, and circular trajectories. These patterns are chosen for their relevance in different scientific domains.

1.1.1. Brownian Motion

Brownian motion, also known as a random walk, models the random movement of particles. This process is defined mathematically by the equation:

$$X(t + \Delta t) = X(t) + \sqrt{2D\Delta t} \cdot N(0, 1)$$

where $X(t)$ is the position at time t, D is the diffusion coefficient, Δt is the time step, and $N(0, 1)$ is a standard normal variable.

1.1.2. Lévy Flight

Lévy flights are characterized by step lengths that follow a heavy-tailed distribution, typically a Lévy distribution. The step length L is given by:

$$L \sim \text{Levy}(\alpha, \beta)$$

where α and β are parameters of the distribution. The trajectory updates similarly to Brownian motion but with step lengths L.

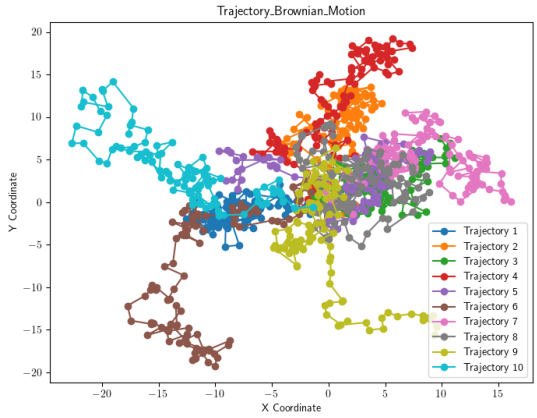

Figure 1: Example of a Brownian motion trajectory

1.1.3. Spiral Trajectory

Spiral trajectories are observed in various systems [1, 2]. The position $(x(t), y(t))$ at time t is given by:

$$x(t) = r(t)\cos(\omega t)$$

$$y(t) = r(t)\sin(\omega t)$$

where $r(t)$ is the radial distance and ω is the angular frequency.

1.1.4. Circular Trajectory

Circular trajectories can be modeled by [1]:

$$x(t) = R\cos(\omega t)$$

$$y(t) = R\sin(\omega t)$$

where R is the radius and ω is the angular frequency.

1.2. Topological Data Analysis (TDA)

1.2.1. Analysis of Spatial Properties

To explore the spatial attributes of trajectory data, we utilize algebraic topology, particularly the Rips complex for geometrical analysis. A simplex in our context is defined as a geometric entity formed by $d+1$ vertices, $v_0, \ldots, v_d$, where each pair of vertices is connected by edges that are linearly independent [3]. In a practical sense, a d-simplex can be visualized as a geometric object that spans d dimensions, such as a point (0-simplex), a line segment (1-simplex), a triangle (2-simplex), and so on [4].

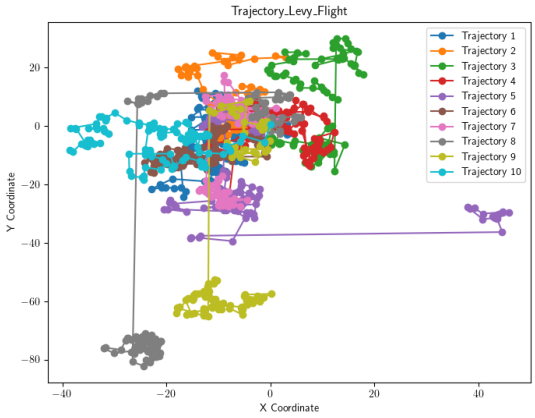

Figure 2: Example of a Lévy flight trajectory

The datasets are composed of positional coordinates given as (x, y), derived from GPS measurements, which record the trajectories within a two-dimensional plane. These coordinates are analyzed collectively as compact trajectories. The construction of the Rips complex, $R_\epsilon(S)$, at a scale ϵ, involves considering all subsets of the trajectory dataset S whose members (points) maintain a maximum pairwise distance no greater than ϵ [5]. As ϵ increases, the complex evolves from a set of isolated vertices (points) to increasingly include higher-dimensional simplices, dependent on the proximity of points [6].
This process transforms the discrete point cloud data into a structured topological space, facilitating the analysis of persistent geometrical features using topological data analysis. Such an approach unveils the underlying spatial structures in the trajectory data, shedding light on the geometric relationships and interactions among the points [7].

1.2.2. Persistence Diagrams

Persistence diagrams are employed to systematically capture the evolution of topological features across various scales in the dataset. These diagrams are crucial for identifying and cataloguing the emergence and dissolution of features within Rips complexes.
A persistence diagram plots the birth and death of topological features as the parameter ϵ varies within our Rips complex construction. Each point (b, d) in the diagram represents a topological feature, with the birth scale b as the x-coordinate, indicating when the feature appears, and the death scale d as the y-coordinate, showing when the feature vanishes [8].
The diagrams provide a concise overview of the data's topological robustness

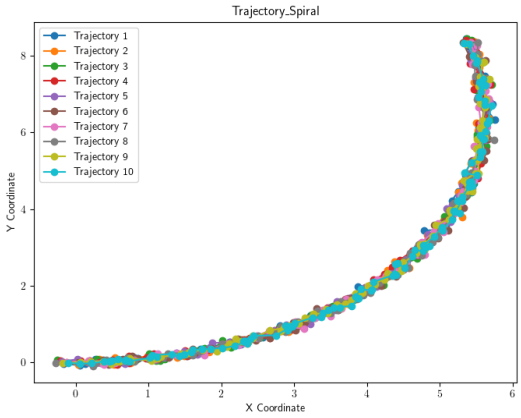

Figure 3: Example of a spiral trajectory

and geometric structure. By distinguishing persistent features from transient noise, we can identify significant shapes and patterns in the data, facilitating a deeper understanding of the underlying spatial relationships [9].

1.2.3. Lifetime of Features

The lifetime L of a topological feature is defined as the difference between its death and birth times:

$$L = d - b$$

This measure provides insight into the persistence and significance of topological features within the data.

1.2.4. Persistence Images

Persistence images convert persistence diagrams into fixed-size vector representations, enabling their integration into machine learning frameworks. The image is computed as:

$$I(x, y) = \sum_{(b,d)\in D} \exp\left(-\frac{(x-b)^2 + (y-(d-b))^2}{2\sigma^2}\right)$$

where σ is a bandwidth parameter that controls the smoothness of the resulting image.

1.2.5. Barycenter of Persistence Diagrams

The analysis of barycentres, or Frechet means, within collections of persistence diagrams is a pivotal method for summarizing and comparing topological features across multiple observations. Barycentres represent the central

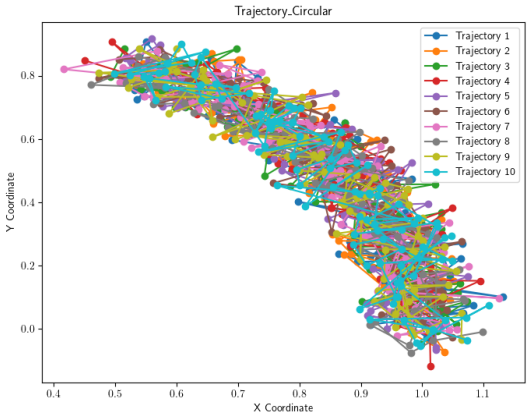

Figure 4: Example of a circular trajectory

tendency of these diagrams and are instrumental in identifying consistent topological patterns.

The barycentre of persistence diagrams is defined as the diagram that minimizes the cumulative Wasserstein distance to all other diagrams in the set. Given a collection of persistence diagrams $\{PD_1, PD_2, \ldots, PD_n\}$, the barycentre μ is computed by solving the following optimization problem:

$$\mu = \arg\min_{\mu} \sum_{i=1}^{n} W_2^2(\mu, PD_i),$$

where $W_2(\mu, PD_i)$ represents the Wasserstein distance between the barycentre μ and each persistence diagram PD_i [10].

The optimization for finding μ typically involves iterative methods that adjust μ by minimizing the overall Wasserstein distance in each iteration. This procedure adjusts the positions and intensities of points in μ to better align with the aggregate features observed across all diagrams, using algorithms such as gradient descent or coordinate descent [11].

Utilizing the barycentre of persistence diagrams is particularly advantageous in comparative analyses where the essence of common features across different datasets is sought. Researchers can use the barycentre to elucidate overarching patterns that are prevalent across various sets of data, enhancing the understanding of the dataset's structural integrity and similarities.

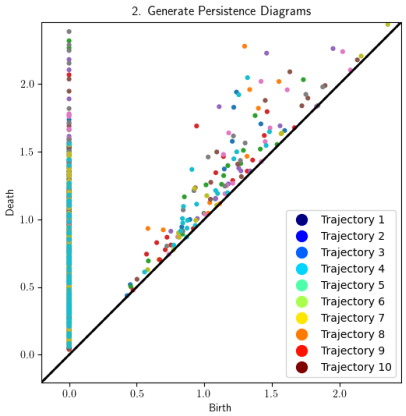

Figure 5: Example of a persistence diagram

1.3. Classification

1.3.1. Logistic Regression

Logistic regression is a statistical method used for binary classification that models the probability of a binary outcome based on one or more predictor variables. The logistic function (sigmoid function) is employed to map predicted values to probabilities:

$$P(Y = 1|X) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n))}$$

where $\beta_0, \beta_1, \ldots, \beta_n$ are the coefficients estimated from the data.
Logistic regression is widely used due to its simplicity and interpretability, particularly in medical and social sciences. It assumes a linear relationship between the log-odds of the outcome and the predictor variables.

1.3.2. Support Vector Machine (SVM)

Support Vector Machines are powerful supervised learning models used for classification and regression tasks. SVMs work by finding the hyperplane that best separates the data points of different classes with the maximum margin. The decision boundary is determined by the support vectors, which are the data points closest to the hyperplane.
The optimization problem for SVM can be formulated as:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{subject to} \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad \forall i$$

where $\mathbf{w}$ is the weight vector and b is the bias term.
SVMs are particularly effective in high-dimensional spaces and are used in applications such as text classification and image recognition.

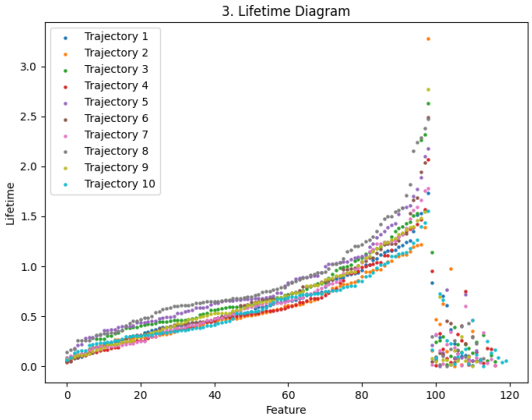

Figure 6: Example of a lifetime diagram

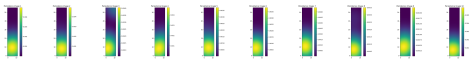

Figure 7: Example of a persistence image

1.3.3. Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes for classification or the mean prediction for regression of the individual trees. It leverages bagging (bootstrap aggregating) to improve the model's robustness and accuracy.

Each tree in a random forest is trained on a random subset of the data, and features are randomly selected at each split. This reduces overfitting and increases generalization. The final prediction is made by averaging the predictions from all individual trees.

Random Forest is widely used in various fields such as finance, healthcare, and ecology for its accuracy and ability to handle large datasets with high dimensionality.

1.3.4. Neural Networks

Neural Networks are a class of machine learning models inspired by the structure and function of the human brain. They consist of layers of interconnected neurons, where each neuron applies a weighted sum of its inputs followed by a non-linear activation function.

The basic unit of a neural network is the perceptron, and the network is trained using backpropagation to minimize the loss function. Neural net-

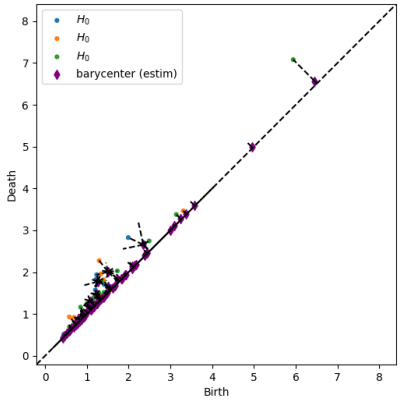

Figure 8: Example of a barycenter calculation

works can capture complex patterns in data and are the foundation of deep learning, which involves multiple hidden layers and large-scale datasets. Neural networks are used in a variety of applications, including image and speech recognition, natural language processing, and autonomous systems.

1.4. Evaluation and Refinement

The evaluation of classification models is essential to assess their performance and generalization capabilities. Common metrics used for evaluating classification models include accuracy, precision, recall and F1 score.

1.4.1. Precision

Precision is the ratio of correctly predicted positive observations to the total predicted positives. It is calculated as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

where TP represents true positives and FP represents false positives. Precision is crucial in scenarios where the cost of false positives is high.

1.4.2. Recall

Recall (or Sensitivity) is the ratio of correctly predicted positive observations to all observations in the actual class. It is calculated as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

where FN represents false negatives. Recall is important in scenarios where the cost of false negatives is high.

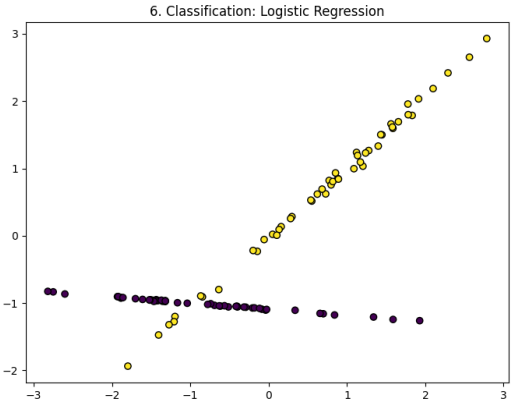

Figure 9: Example of a logistic regression decision boundary

1.4.3. Accuracy

Accuracy is the ratio of correctly predicted instances (both true positives and true negatives) to the total number of instances. It is calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where TN represents true negatives. Accuracy provides a general measure of model performance, but it may be misleading in the case of imbalanced datasets.

1.4.4. F1-Score

The F1-score is the harmonic mean of precision and recall. It provides a single metric that balances both precision and recall:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1-score is particularly useful when the class distribution is imbalanced and a balance between precision and recall is desired.

References

[1] E. Lauga, T. R. Powers, The hydrodynamics of swimming microorganisms, Reports on Progress in Physics 72 (9) (2009) 096601.

[2] M. J. Lighthill, Mathematical Biofluiddynamics, SIAM, 1975.

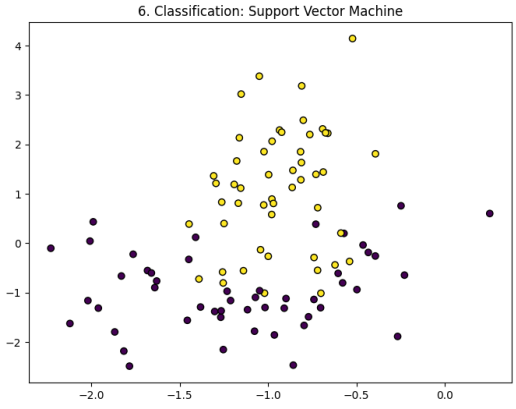

Figure 10: Example of an SVM decision boundary

[3] A. Hatcher, Algebraic topology, Cambridge University Press, 2002.

[4] J. R. Munkres, Elements of Algebraic Topology, Addison-Wesley, 1984.

[5] R. Ghrist, Elementary Applied Topology, Createspace, 2014.

[6] G. Carlsson, Topology and data, Bulletin of the American Mathematical Society 46 (2) (2009) 255–308.

[7] H. Edelsbrunner, D. Letscher, A. Zomorodian, Topological persistence and simplification, Discrete & Computational Geometry 28 (4) (2000) 511–533.

[8] H. Edelsbrunner, J. Harer, Computational topology: an introduction, American Mathematical Soc., 2010.

[9] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, Stability of persistence diagrams, Discrete & Computational Geometry 37 (1) (2007) 103–120.

[10] K. Turner, Y. Mileyko, S. Mukherjee, J. Harer, Fréchet means for distributions of persistence diagrams, Discrete & Computational Geometry 52 (1) (2014) 44–70.

[11] V. Divol, T. Lacombe, Understanding the topology and the geometry of the space of persistence diagrams via optimal partial transport, Journal of Applied and Computational Topology 5 (2021) 1–53.

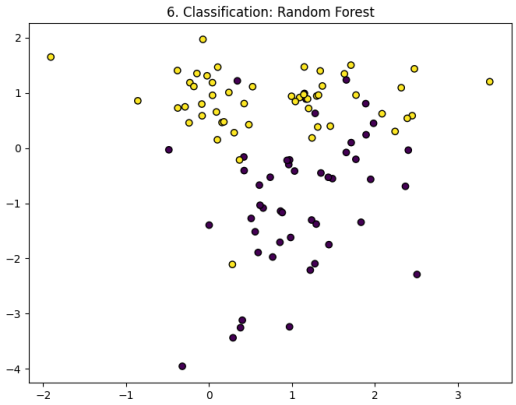

Figure 11: Example of a random forest structure

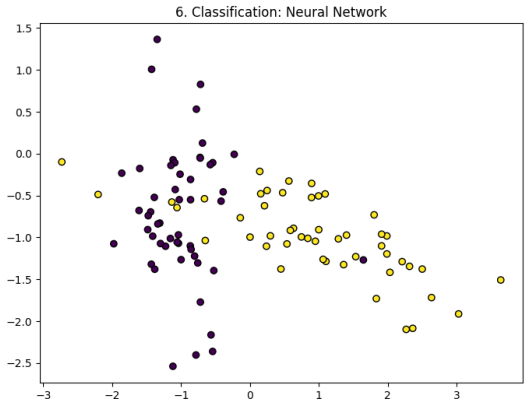

Figure 12: Example of a neural network architecture

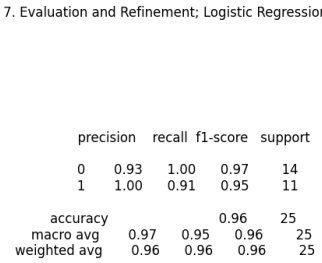

Figure 13: Example of classification metrics in a Logistic Regression model