# TDA Pipeline

Rok Filipovič, Rok Mušič

January 2025

**Abstract**

The following paper presents our project in Topological Data Analysis (TDA) course. It explains each step required for a proper TDA pipeline, namely data acquisition and preprocessing, filtration, persistence homology, vectorization, and, lastly, analysis of the final results. In our case, vectorization is performed using persistence landscapes.

We test our implementation on data from Our World in Data website, mainly surrounding $CO_2$ emissions [5].

## 1 Introduction

Given some point-cloud, our task is to follow the TDA pipeline to obtain a better understanding of our original points. We start by preprocessing data into a suitable format for our custom filtration technique and using the latter to acquire persistence homology for each individual point. Due to the nature of our filtration, each point has a different persistence homology, however, those that are part of a similar structure, obtain a similar one. This characteristic allows us to group points that are topologically similar and cannot be clustered using other methods. To compare the persistence homologies, however, we first embed them into a Euclidean space, where we can compare them in a typical manner. This is done by vectorizing persistence landscapes obtained from persistence homologies.

## 2 Methods

In this section we discuss in depth, each step in the TDA Pipeline. We start with filtration to obtain persistence homologies, which are turned into vectors in the Euclidean world.

### 2.1 Filtration and persistence homology

Our filtration is motivated by the one used in Stable Topological Signatures for Points on 3D Shapes, where they obtained distinct persistence homologies for

each point on the surface of a hand [2]. In our work we, however, generalize this approach to any set of points.

Every point in our dataset is treated as a graph node, and edges connect nodes that are below a specified distance threshold. This threshold is determined empirically using principal component analysis (PCA). However, the resulting graph may contain multiple components. As our algorithm requires a connected graph, we connect distinct components by adding an edge between the current two closest points that are not a part of a connected component. If the threshold edges result in $k$ components with $n$ edges, we are left with 1 component and $n + (k - 1)$ edges.

The basis of our filtration lies in breath-first-search (BFS) over this graph. It is important to note that starting BFS in a different node of the graph yields a different result. We use this fact to obtain a different persistence homology for each point of our dataset. Despite this, we hope that nearby points and those points that are a part of similar structures have similar BFS results, therefore similar filtrations and lastly similar persistence homologies.

Every point is added to the filtration in step 0. At step $i$ we then add an edge between a parent and a child in the tree obtained by BFS, if and only if the child is $i$ hops away from the origin. Additionally, since BFS produces a tree, we add an edge between two points if they are neighboring and are both already a part of a 1-dimensional simplex present in the filtration. We do so, to close up gaps. Here we can notice, that an edge between two neighboring vertices $u$ and $v$ always appears at step $i$ or $i + 1$, where $i = min(\#$ hops to $u, \#$ hops to $v)$.

Lastly, we add a 2-dimensional simplex to the filtration at step $i$, if three vertices form a clique based on the edges present in the filtration. We could also delve deeper using this approach and try to work with even more dimensional simplices, but it turns out that is not very beneficial.

Building filtration for each point with steps from above achieves the task of obtaining distinct persistence homologies for every individual point. We can therefore transform these persistence homologies into vectors to interpret them as elements of the Euclidean space.

## 2.2 Vectorization

As mentioned in the abstract, we used persistence landscapes [1] as our vectorization technique.

But first let us take a step back and elaborate on why even bother with vectorization. After all, we are doing topological data analysis, and persistence diagrams conveniently tell us all the important topological properties of the data. However, "classical" machine learning techniques don't speak topology, but linear algebra (and statistics, and calculus, and you get the point). So in order to be able to utilize such a powerful toolkit, we need to be able to represent the data in a suitable format.

The concept of persistence landscapes is relatively straightforward. The input to the method is a persistent diagram consisting of $(birth, death)$ pairs

and transform them into tent like functions. We map each $(b, d)$ interval into a piecewise function:

$$f_{b,d}(x) = \begin{cases} 0 & \text{if } x < b \text{ or } x > d, \\ x - b & \text{if } b \leq x \leq \frac{b+d}{2}, \\ d - x & \text{if } \frac{b+d}{2} < x \leq d. \end{cases}$$

We define the function to be 0 everywhere outside the $(birth, death)$ interval. Starting at $x = b$, the function starts linearly rising to $x = \frac{b+d}{2}$, which is the middle point of the interval. In other words, $f_{b,d}(x)$ behaves the same as $f(x) = y$, if we translate the coordinate system by $(b, 0)$ (move it $b$ units to the right, so that $b$ would be the new center of coordinate system). After the middle point, the function decreases linearly toward the point $(d, 0)$. In a similar fashion, we can compare this decreasing function to $f(x) = -y$ for an appropriately translated coordinate system.

We do this for every interval in the persistent diagram, thus obtaining many different piecewise functions. The next step is to determine the sampling range and the resolution.

For the sampling range we simply take the interval $(b_{min}, d_{max})$ where

$$b_{min} = \min_{b \in PD} b$$
$$d_{max} = \max_{d \in PD} d$$

The persistence diagram is abbreviated as PD, and the values are, of course, meant to be interpreted appropriately (we hope the reader can forgive us for such an abuse of notation).

There are two special cases when it comes to determining $d_{max}$:

1. If any of the death values are infinite, we set them as max finite death + 1. We experimented with setting different values for this, for example $2 \cdot$ max finite death and others, but ultimately decided to stick with adding 1. The main reason for this is that when we split the interval (explained later), we are left with a lot more meaningful values (recall that $f_{b,d}(x) = 0$ for $x$ outside the interval, so by setting the max death to a large value, a lot of piecewise functions will be 0 for the majority of the time, losing valuable information).

2. If all death values are infinite, we set them as max birth + 1. The reasoning is as above.

The sampling range is now split into $n$ evenly sized intervals, where $n$ is the resolution. Setting a high resolution results in functions being evaluated at more points and getting more distinct and accurate measurements, but results in the vectors being very high-dimensional. In an opposite manner, choosing a low resolution means the functions are evaluated sparsely, which could result in

distorted information. We tried different values to try to balance one with the other, with the values of resolution ranging from 2 and up to 100. In the end, $resolution = 10$ worked well in our case.

Now we arrive at what a landscape even is. Thus far we have constructed the piecewise functions, determined the sampling range and the sampling values, and evaluated the functions at all sampled values. The first persistent landscape is obtained by taking a maximum value of all piecewise functions at every sampled value. The second landscape is obtained by taking the second highest values, and so on.

In this manner, if we choose the first $k$ landscapes, we get $k$ vectors, one for each of the landscape. This technically leaves us with a matrix. We tried two approaches of aggregating the vectors into one final vector:

1. Aggregating them together by taking a mean of the vectors.
2. Concatenating the vectors into one vector. This one produced better results. We hypothesize that is because all information is retained.

The final step was to concatenate the vectors from different homology groups. That is, a vector was calculated for each $H_i$ in the persistence diagram, and all those vectors were concatenated at the end. In the end, we get a single final vector for each country.

# 3 Results on $CO_2$ Dataset

We now put these methods to use. We start by presenting the dataset itself and then explain what each method from Section 2 produces. Lastly, we enterpret the obtained results.

## 3.1 Dataset

We try out the implementation on data from Our World in Data website. We focus on data regarding $CO_2$ emissions [5], so we extract features that are relevant in that regard. Besides $CO_2$ consumption we obtain newest available data for the following topics for each country: Percentage of adults with at least some basic education [10], average years of formal education for adults [9], electricity generation from solar and wind per capita [6], greenhouse gas emissions per capita [7], energy use per capita [8], GDP per capita [11], inequality index [4] and lastly, life expectancy [3]. Additionally, all data is scaled to 0-1, to ensure that some features do not override others due to scale. Lastly, for future work we only consider countries that had an existing value at every feature. We are left off with 123 rows, each presenting one country, each containing 9 features.

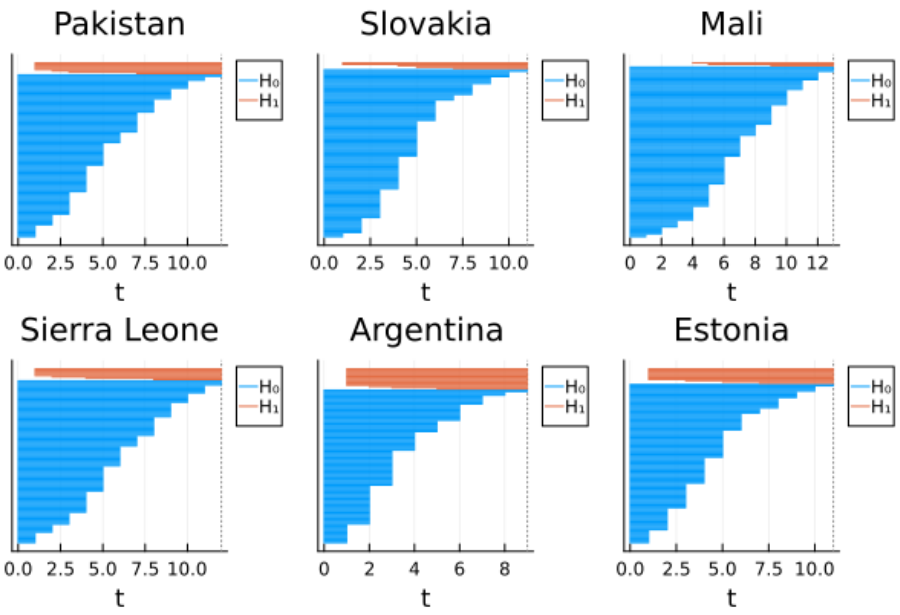


Figure 1: Bar-code diagrams of persistence diagrams for 6 countries obtained using our custom filtration. We can notice similarities and differences in $H_0$ and $H_1$ bars, both in shape and in life span.

## 3.2 Results

### 3.2.1 Filtration and persistence homology

We build a graph by adding an edge between every pair of nodes that are at a distance of $\epsilon = 0.25$ or less. Doing so, we are left with 16 connected components, most of them containing just a single vertex. We therefore settle for such $\epsilon$ and connect these points to the nearest other vertex.

After building a filtration and running `ripserer()` function for each individual country, we are left with distinct persistence homologies. Figure 1 contains bar-code diagrams of a few obtained examples. We can notice that Pakistan and Estonia have very similar bar codes in 0-dimensional and 1-dimensional persistence diagrams. In addition, Mali differs from all other five, as some of its $H_0$ bars die out as late as in 12th iteration. On top of that, its $H_1$ persistence diagram only contains 3 elements, while all others have at least 5.

### 3.2.2 Clustering

We used DBSCAN algorithm to cluster the vectors obtained after the vectorization step. The clustering algorithm is suitable for our task, as it does not require a predetermined number of clusters, is not dependent on the underlying space (which is assumed to be Euclidean space) like K-means clustering, and

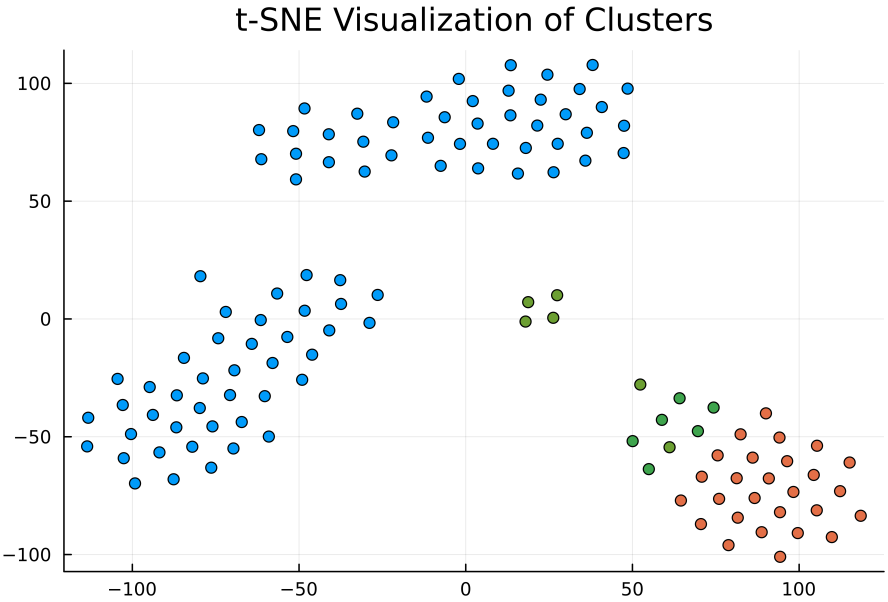


Figure 2: A TSNE visualization of the clusters obtained using the DBSCAN algorithm.

resilient to outliers.

The resulting clustering, after trying different algorithm configurations, can be seen in Figure 2.

The algorithm found 4 clusters, which are nicely visualized using the TSNE algorithm. Now let us turn our attention to the original data and try to infer some insides based on this clustering.

### 3.2.3 Interpretation of Results

Table 3.2.2 shows clear differences in how each cluster behaves according to the original features. Below is a more detailed discussion of these findings:

- **Cluster 2** displays the highest $CO_2$ ($0.246 \pm 0.244$), *EnergyUse* ($0.283 \pm 0.284$), and *GreenhouseGasEmissions* ($0.201 \pm 0.230$). Although it has a relatively high *GDP* ($0.244 \pm 0.239$), its *FormalEducationPercent* ($0.761 \pm 0.315$) is noticeably lower than the other clusters. This pattern suggests that countries in Cluster 2 are likely industrialized or heavily dependent on fossil fuels, contributing to elevated emission levels.

- **Cluster 3** stands out for its combination of high *GDP* ($0.259 \pm 0.141$), high *FormalEducationPercent* ($0.949 \pm 0.088$), and the greatest reliance on *SolarAndWindElectricity* ($0.395 \pm 0.333$). Despite having a robust economy (comparable or even higher GDP than Cluster 2), it maintains a

| Feature | Cluster 0 | Cluster 1 | Cluster 2 | Cluster 3 |
|---|---|---|---|---|
| CO2 | $0.121 \pm 0.1$ | $0.069 \pm 0.056$ | $0.246 \pm 0.244$ | $0.097 \pm 0.036$ |
| FormalEducationPercent | $0.801 \pm 0.255$ | $0.864 \pm 0.194$ | $0.761 \pm 0.315$ | $0.949 \pm 0.088$ |
| EnergyUse | $0.164 \pm 0.131$ | $0.074 \pm 0.065$ | $0.283 \pm 0.284$ | $0.137 \pm 0.066$ |
| GDP | $0.144 \pm 0.103$ | $0.105 \pm 0.09$ | $0.244 \pm 0.239$ | $0.259 \pm 0.141$ |
| Inequality | $0.551 \pm 0.335$ | $0.587 \pm 0.192$ | $0.581 \pm 0.251$ | $0.512 \pm 0.197$ |
| LifeExpectancy | $0.807 \pm 0.141$ | $0.751 \pm 0.131$ | $0.776 \pm 0.174$ | $0.86 \pm 0.093$ |
| SolarAndWindElectricity | $0.086 \pm 0.104$ | $0.112 \pm 0.177$ | $0.158 \pm 0.275$ | $0.395 \pm 0.333$ |
| GreenhouseGasEmissions | $0.084 \pm 0.071$ | $0.05 \pm 0.039$ | $0.201 \pm 0.23$ | $0.076 \pm 0.02$ |
| SchoolingYears | $0.58 \pm 0.264$ | $0.551 \pm 0.211$ | $0.556 \pm 0.303$ | $0.698 \pm 0.279$ |

Table 1: The mean distribution and standard deviation of the original features, grouped by cluster. The clusters have their colors assigned from left to right in the following order: green, blue, orange, dark green (see TSNE visulization in Figure 2).

moderate $CO_2$ level ($0.097 \pm 0.036$). The elevated *SchoolingYears* ($0.698 \pm 0.279$) and the highest *LifeExpectancy* ($0.860 \pm 0.093$) further hint that better education and broader adoption of clean energy sources may play a significant role in keeping emissions lower.

- **Cluster 1** has the lowest mean $CO_2$ ($0.069 \pm 0.056$) and *EnergyUse* ($0.074 \pm 0.065$), along with a relatively low *GDP* ($0.105 \pm 0.090$). The smaller environmental footprint could stem from less industrial activity. Interestingly, *FormalEducationPercent* ($0.864 \pm 0.194$) remains relatively high, which indicates that these countries might have good educational coverage but lower energy demands overall.

- **Cluster 0** holds a middle ground in almost all features. Its $CO_2$ ($0.121 \pm 0.100$), *EnergyUse* ($0.164 \pm 0.131$), and *GDP* ($0.144 \pm 0.103$) fall between those of Clusters 1 and 2/3. Similar moderation is seen for *LifeExpectancy* ($0.807 \pm 0.141$) and *SchoolingYears* ($0.580 \pm 0.264$). These observations could point to countries in transitional stages of development, exhibiting moderate levels of both industrialization and educational attainment.

In summary, the cluster analysis indicates that higher $CO_2$ emissions are closely tied to elevated *EnergyUse* and reliance on non-renewables (Cluster 2), whereas clusters with more focus on clean energy sources (Cluster 3) can maintain a balance between economic development and lower emissions. Clusters 0 and 1 sit at different points between these two extremes, one with moderate traits (Cluster 0) and the other with lower industrial activity and emissions (Cluster 1).

## Division of work

Rok Filipovič obtained the persistence homology of each point in the dataset, while Rok Mušič created and vectorized the landscapes from them and reasoned about the final results.

## References


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

[2] Mathieu Carrière, Steve Y. Oudot, and Maks Ovsjanikov. Stable topological signatures for points on 3d shapes. *Computer Graphics Forum*, 34(5):1–12, 2015.

[3] Saloni Dattani, Lucas Rodés-Guirao, Hannah Ritchie, Esteban Ortiz-Ospina, and Max Roser. Life expectancy. *Our World in Data*, 2023. Online resource, accessed on 2024-12-31.

[4] Joe Hasell, Pablo Arriagada, Esteban Ortiz-Ospina, and Max Roser. Economic inequality. *Our World in Data*, 2023. Online resource, accessed on 2024-12-31.

[5] Hannah Ritchie, Pablo Rosado, and Max Roser. Co and greenhouse gas emissions. *Our World in Data*, 2023. Online resource, accessed on 2024-12-31.

[6] Hannah Ritchie, Pablo Rosado, and Max Roser. Data page: Electricity generation from solar and wind power per person. Part of the publication: Energy. Data adapted from Ember, Energy Institute, Various sources, 2023. Online resource, accessed on 2024-12-31.

[7] Hannah Ritchie, Pablo Rosado, and Max Roser. Data page: Per capita greenhouse gas emissions including land use. Part of the publication: CO and Greenhouse Gas Emissions. Data adapted from Jones et al., Various sources, 2023. Online resource, accessed on 2024-12-31.

[8] Hannah Ritchie, Pablo Rosado, and Max Roser. Data page: Primary energy consumption per capita. Part of the publication: Energy. Data adapted from U.S. Energy Information Administration, Energy Institute, Various sources, 2023. Online resource, accessed on 2024-12-31.

[9] Hannah Ritchie, Veronika Samborska, Natasha Ahuja, Esteban Ortiz-Ospina, and Max Roser. Data page: Average years of schooling. Part of the publication: Global Education. Data adapted from Barro and Lee, Lee and Lee, 2023. Online resource, accessed on 2024-12-31.

[10] Hannah Ritchie, Veronika Samborska, Natasha Ahuja, Esteban Ortiz-Ospina, and Max Roser. Data page: Population having attained at least some formal education. Part of the publication: Global Education. Data adapted from Barro and Lee, Lee and Lee, 2023. Online resource, accessed on 2024-12-31.

[11] Max Roser, Pablo Arriagada, Joe Hasell, Hannah Ritchie, and Esteban Ortiz-Ospina. Data page: Gdp per capita. Part of the publication: Economic Growth. Data adapted from World Bank, 2023. Online resource, accessed on 2024-12-31.