Complete Beginner's Guide to Topological Data Analysis: A Step-by-Step Walkthrough

Ali Al-Barkawi, Montader Alasady

Abstract

This guide provides a detailed, beginner-friendly walkthrough of topological data analysis using persistent homology and the Mapper algorithm. Designed for readers with no prior background in topology or TDA, each step includes explicit instructions for hand computation, visual aids, and clear interpretations. Using six points on a circle, readers will learn to detect topological features and construct meaningful graph summaries of data structure.

1 What You'll Learn

You'll discover how to find "shapes" hidden in data using two powerful techniques:

1. **Persistent Homology**: Detects holes and loops in your data
2. **Mapper Algorithm**: Creates a network diagram that shows the data's structure

No prior knowledge needed! Just grab paper, pencil, and a calculator.

2 Part 1: Understanding Your Data

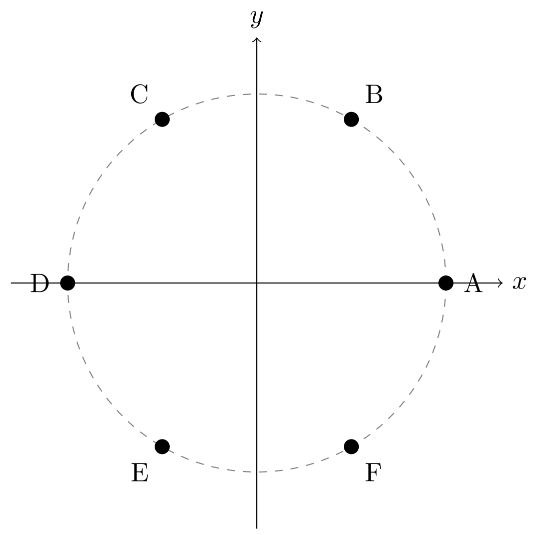
2.1 Step 1: Plot Your Points

You have 6 points arranged in a perfect circle. The coordinates are:

$$\begin{aligned} A &= (1, 0), \\ B &= (0.5, 0.87), \\ C &= (-0.5, 0.87), \\ D &= (-1, 0), \\ E &= (-0.5, -0.87), \\ F &= (0.5, -0.87). \end{aligned}$$

What to do:

- Draw an X-Y coordinate plane on graph paper
- Mark each point at its location
- Label them A through F
- Notice they form a circle!



2.2 Step 2: Calculate Distances Between Points

We need to know how far apart each pair of points is.
What to do:

1. Use the distance formula: $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$
2. Or use this shortcut since they're on a circle:
   - **Neighbors** (like A to B): distance = 1
   - **Skip one** (like A to C): distance = 1.73 ($\sqrt{3}$)
   - **Across the circle** (like A to D): distance = 2

Create this distance table:

	A	B	C	D	E	F
A	0	1	1.73	2	1.73	1
B	1	0	1	1.73	1.73	1.73
C	1.73	1	0	1	1.73	1.73
D	2	1.73	1	0	1	1.73
E	1.73	1.73	1.73	1	0	1
F	1	1.73	1.73	1.73	1	0

3 Part 2: Persistent Homology (Finding Loops)

3.1 What Are We Doing?

Imagine slowly inflating a balloon around each point. As balloons touch, we connect the points. We're watching for when loops form and when they disappear.

3.2 Step 3: Build the Filtration at $\varepsilon = 0$

ε (epsilon) is our "balloon radius."
What to do:

1. Draw your 6 points on paper with NO lines between them
2. Count connected groups: **6 separate points**

3. Count loops: **0 loops**

Record: At $\varepsilon = 0$:

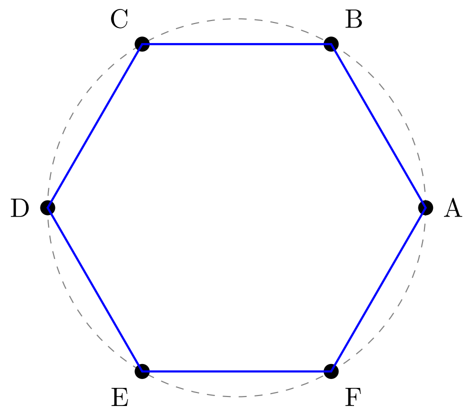
- H_0 (connected components) = 6
- H_1 (loops) = 0

3.3 Step 4: Build the Filtration at $\varepsilon = 1$

Now connect any points that are distance ≤ 1 apart.
What to do:

1. Look at your distance table
2. Draw lines between all pairs with distance = 1:
   - Connect A–B (distance 1) ✓
   - Connect B–C (distance 1) ✓
   - Connect C–D (distance 1) ✓
   - Connect D–E (distance 1) ✓
   - Connect E–F (distance 1) ✓
   - Connect F–A (distance 1) ✓

You should now see a hexagon!



3. Count connected groups: **1 big group** (all points connected)
4. Count loops: **1 loop** (the hexagon goes around)

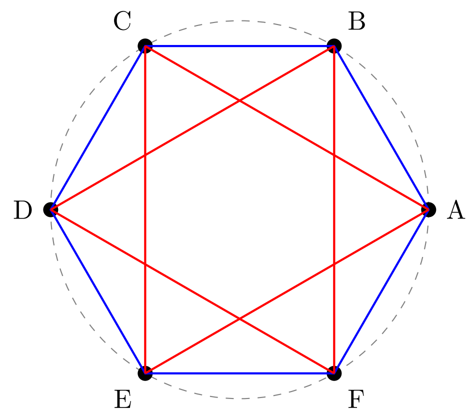
Record: At $\varepsilon = 1$:

- $H_0 = 1$ (all merged into one component)
- $H_1 = 1$ (one loop was **BORN!**)

3.4 Step 5: Build the Filtration at $\varepsilon = 1.73$ ($\sqrt{3}$)

Now connect points that are distance ≤ 1.73 apart.
What to do:

1. Keep all your previous lines (distance 1)
2. Add new lines for distance 1.73:
   - A–C, A–E
   - B–D, B–F
   - C–E
   - D–F
3. Look for **triangles** (filled-in areas):
   - Triangle A-B-C: Check if all three edges exist
     - A–B ✓ (distance 1)
     - B–C ✓ (distance 1)
     - A–C ✓ (distance 1.73)
     - **This is a filled triangle!**
4. When you fill in triangles, the loop gets "plugged up"



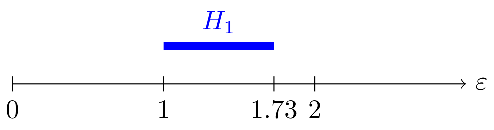
Record: At $\varepsilon = 1.73$:

- $H_0 = 1$ (still one component)
- $H_1 = 0$ (the loop **DIED**—it got filled in!)

3.5 Step 6: Create Your Barcode

A barcode shows how long each feature "lives."
What to do:

1. Draw a number line from 0 to 2
2. For the loop (H_1):
   - It was born at $\varepsilon = 1$
   - It died at $\varepsilon = 1.73$
   - Draw a thick bar from 1 to 1.73



Interpretation: The long bar means there's a **significant circular structure** in your data!

4 Part 3: The Mapper Algorithm (Building a Network)

4.1 What Are We Doing?

We'll create a simplified graph that captures the circular shape of our data using a "lens function" to look at the data from a specific angle.

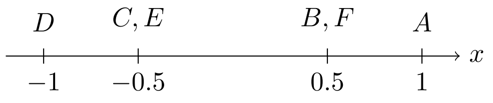
4.2 Step 7: Choose Your Lens Function

A lens is just picking one measurement from each point. We'll use the x-coordinate.
What to do:

1. Write down each point's x-value:

$$A : x = 1.0 \qquad B : x = 0.5 \qquad C : x = -0.5$$
$$D : x = -1.0 \qquad E : x = -0.5 \qquad F : x = 0.5$$

2. Draw a number line from -1 to 1
3. Mark where each point falls on this line



4.3 Step 8: Create Overlapping Intervals

Divide the number line into 3 overlapping sections (like overlapping windows).
What to do:

1. Draw three intervals:
   - I_1: $[-1.0, -0.2]$ (left section)
   - I_2: $[-0.6, 0.6]$ (middle section, overlaps with both)
   - I_3: $[0.2, 1.0]$ (right section)
2. Assign points to intervals based on their x-values:
   - I_1 contains: $D(-1.0)$, $C(-0.5)$, $E(-0.5)$
   - I_2 contains: $C(-0.5)$, $E(-0.5)$, $B(0.5)$, $F(0.5)$
   - I_3 contains: $B(0.5)$, $F(0.5)$, $A(1.0)$

Notice: Some points appear in multiple intervals (this overlap is crucial!)

4.4 Step 9: Cluster Within Each Interval

Group together points that are close to each other (within distance 1.1).
What to do for $I_1 = \{C, D, E\}$:

1. Check distances:
   - C–D: 1 ✓ (close enough)
   - D–E: 1 ✓ (close enough)
   - C–E: 1.73 × (too far)
2. But since C connects to D, and D connects to E, they're all in one chain
3. **Create cluster** $N_1 = \{C, D, E\}$

What to do for $I_2 = \{B, C, E, F\}$:

1. Check distances:
   - B–C: 1 ✓ → Group B and C
   - E–F: 1 ✓ → Group E and F
   - B and F are distance 1.73 (too far to merge)
2. **Create two clusters:**
   - $N_2 = \{B, C\}$
   - $N_3 = \{E, F\}$

What to do for $I_3 = \{A, B, F\}$:

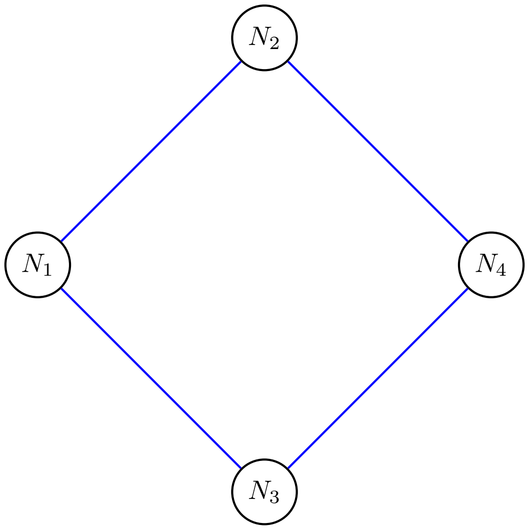
1. Check distances:
   - A–B: 1 ✓
   - A–F: 1 ✓
   - B–F: 1.73 (just slightly too far, but A connects both)
2. They form a connected chain
3. **Create cluster** $N_4 = \{A, B, F\}$

4.5 Step 10: Build the Mapper Graph

Connect clusters that share any points.
What to do:

1. Draw 4 nodes (circles) labeled N_1, N_2, N_3, N_4
2. Check which clusters share points:
   - N_1 contains C, N_2 contains C → **Connect** N_1–N_2
   - N_1 contains E, N_3 contains E → **Connect** N_1–N_3
   - N_2 contains B, N_4 contains B → **Connect** N_2–N_4
   - N_3 contains F, N_4 contains F → **Connect** N_3–N_4

Your final graph:



This forms a square/4-cycle that captures the circular structure!

5 Part 4: What Did We Discover?

5.1 From Persistent Homology:

- The barcode showed one strong loop existing from $\varepsilon = 1$ to $\varepsilon = 1.73$
- This tells us: **"The data has a circular structure!"**

5.2 From Mapper:

- The graph formed a 4-node cycle
- This also reveals: **"The data wraps around in a circle!"**

5.3 Why This Matters:

Both methods independently discovered that our 6 points form a circle, even though we never explicitly told the algorithms about circles! This is the power of topological data analysis—it finds **shapes** in data automatically.

6 Try It Yourself!

1. **Exercise 1:** What happens if you use $\varepsilon = 2.5$ in the persistent homology? *Hint: Check your distance table—what new connections form?*
2. **Exercise 2:** What if you used the y-coordinate as your lens function instead of x?
3. **Exercise 3:** Try this with your own 4–6 points arranged in a different pattern!

7 Key Vocabulary

Filtration: Gradually connecting points as we increase ε

H_0: Number of separate clusters

H_1: Number of loops/holes

Born: When a loop first appears

Died: When a loop gets filled in

Persistence: How long a feature lasts (death − birth)

Lens function: A way to “project” or view your data

Cover: Overlapping intervals that span the lens range