Rutgers University

Course Syllabus for MATH495

Introduction to Topological Data Analysis

Spring 2026

General Information

Instructor	Mariano Echeverria	Meeting:	*** ***
Email	mariano.echeverria@rutgers.edu		

Important Information

Prerequisites

Introduction to Linear Algebra (Math 250) and Multivariable Calculus (Math 251) or equivalent courses. In case you don't meet one of the two requirements, reach out to me for advice and fill out the Prerequisite Override Form for assistance with registration.

https://rutgers.ca1.qualtrics.com/jfe/form/SV_9H9FGkA49Z7BpBk

Also, since this is a topics course, being interested in these topics is equally important.

What is Topological Data Analysis (TDA) for?

Topology is one of the most powerful branches of Mathematics, focused on characterizing different shapes by properties that persist even after the shape has been continuously deformed (hence the joke that a Topologist is someone who can't tell the difference between a coffee cup and a doughnut! wiki:mug-torus).

One of the more recent applications of Topology is Topological Data Analysis (TDA), which is a collection of methods for extracting, analyzing, and visualizing the shape or structure of complex data.

This complex data is usually given as a point cloud in some high-dimensional space. Many times this data has some underlying shape, which conveys useful information: think about points being sampled from a sphere or a torus (which is a fancy name for a doughnut).

In TDA we use the classical tools and language from Topology to detect and describe the notion of "shape" in data sets, and for example, allow us to determine whether our point cloud was generated from a sphere or a torus.

Other questions TDA help answer and applications include:

- How do you numerically find periodic orbits in a complex dynamical system?
- Detecting stable features in the folding of proteins and RNA.
- Persistent homology provides topological summaries (persistence diagrams, barcodes) that help determine robust features for machine learning models.
- Mapper algorithm reveals clusters and geometric structures that traditional linear methods like PCA may miss.

Main Topics (not covered necessarily in this order)

1. Basics of Topology: topological spaces, metric space topology. Homeomorphisms and topological manifolds.
2. Complexes on data: Simplicial complexes, Graph induced complexes.
3. Homology: Chains, boundaries, homology groups, Betti numbers. Induced maps among homology groups. Relative homology groups
4. Topological persistence: Filtrations, Persistent homology, Persistence algorithm, Persistence diagram
5. Topology inference from data: Computing homology from data. Sparsification to handle big data. The Mapper algorithm.
6. Persistence on graphs and Reeb graphs: Reeb graphs, Interleaving distance, comparing graphs with persistence summary.
7. Discrete Morse Theory. Noise filtering in persistence diagrams via Morse cancellations. Persistent Morse-Smale complexes.
8. Some software packages: R for TDA, KeplerMapper.

References

The official textbook is

- Topological Data Analysis with Applications, Gunnar Carlsson and Mikael Vejdemo-Johansson. Cambridge University Press, 2021. Online ISBN: 9781108975704

You don't need to purchase the textbook, since Rutgers Libraries has online access to the textbook (this is one of the reasons the book is the "official" one).

At the same time, will also draw on the following references as needed:

- Elementary Applied Topology, Robert Ghrist.
- Computational Topology and Data Analysis, T. K. Dey and Y. Wang, Cambridge U. Press.
- Computational Topology, Herbert Edelsbrunner and John L. Harer, AMS.

- Topological Data Analysis for Genomics and Evolution: Topology in Biology, Raul Rabdan and Andrew Blumberg.
- Mathematical Principles of Topological and Geometric Data Analysis, Parvaneh Joharinad and Jurgen Jost.
- Geometric and Topological Inference, Jean-Daniel Boissonnat

and for certain application some research articles will be used as well.

Grading Criteria and Class Structure

There won't be any exams on this course. In exchange, we will have (mostly) weekly assignments, some which will require using software packages like R.

Even though most meetings will consist of lectures, hopefully there will be plenty of interactions between all of us to make the class feel more active than a traditional course.