Topological Data Analysis and Persistence Theory

Peter Bubenik

Outline

Lectures

Lecture 1: Motivation and Basic Constructions. We will start with motivations for topological data analysis (TDA) and an overview of TDA. I will introduce the some of the basic mathematical objects and constructions of TDA and discuss the corresponding categories and functors. I will end with an introduction to persistent homology.

Lecture 2: Foundational Results. In the second lecture, we will study the persistence algorithm, Wasserstein distances, interleaving, the isometry theorem, sublevelset stability, Wasserstein stability, and various flavors of persistence.

Lecture 3: Combinatorics and Geometry. In the third lecture, we will see how a combinatorial construction, Möbius inversion, produces persistence diagrams and graded persistence diagrams. I will present the Gromov-Hausdorff stability of persistent homology and introduce generalized persistence diagrams.

Lecture 4: TDA and Statistics. In the fourth lecture, we will discuss feature maps and kernels for TDA. I will use the erosion of persistence modules to define the persistence landscape and discuss some of its properties. I will show how TDA may be used for statistics and exploratory data analysis and discuss dimensionality reduction and convergence results.

Lecture 5: The Algebra of Persistence Modules. In the fifth lecture, we will consider persistence modules as representations of quivers and as graded modules and discuss the classification theorem of persistence modules. We will study maps of persistence modules and the algebraic stability theorem. I will introduce q-tame persistence modules and also consider the algebra of generalized persistence modules.

Lecture 6: TDA and Machine Learning. In the sixth lecture, I will discuss learning problems, clustering, classification, regression, and deep learning, and their connections to TDA.

Lecture 7: Noise and multiparameter persistence. In the seventh lecture, I will discuss noise in data and the resulting instability of persistent homology and how it may be addressed. This will lead us to multiparameter persistent homology (MPH). I will discuss the mathematics of MPH and computational approaches to MPH.

Lecture 8: Applications of TDA. I will present case studies that illustrate a number of important topics in TDA: how long and short bars reveal topology and geometry; the importance of preprocessing and how to represent data as a filtered simplicial complex; how to apply TDA to time series data; and the use of representative cycles in TDA.

Lecture 9: Mathematics and Persistence. In the second-last lecture, I will discuss some interesting mathematical connections to persistence that have arisen in recent research: formal sums on metric spaces; optimal transportation; generalized Morse theory; and abelian categories.

Lecture 10: Software and Algorithms. In the final lecture, I will discuss algorithmic improvements to persistence computations and various persistent homology and TDA software.

Group Work and Discussion

Group Work 1. These exercises with give you practice with the basic constructions of TDA as well as computing homology and the ranks of persistent homology vector spaces.

Group Work 2. These exercises will lead you through the computation of barcodes and persistence diagrams via matrix reduction and via Möbius inversion. You will also compute graded persistence diagrams.

Group Work 3. These exercises will lead you through the computation of the persistence landscape.

Lab Sessions

Lab 1. You will learn how to sample points, construct Voronoi cells, Delaunay complexes and Vietoris-Rips complexes. You will compute barcodes and persistence diagrams, and visualize representative cycles.

Lab 2. You will compute death vectors and persistence landscapes, and learn how to average them, plot their differences, and compute p values. You will perform principal components analysis (PCA) and use it to plot low-dimensional projections, loading vectors, and the explained variance.

Lab 3. You will combine TDA with support vector machine (SVM) to classify data and to perform regression. You will clean noisy data using k-nearest neighbors (kNN).