# EECE 8095 — TOPICS IN COMPUTER SYSTEMS
## Spring 2019: Approximate Methods to Compute Persistent Homology

Tues/Thurs: 3:30-4:50
Room: TBD

**Catalog Data:** 20-260-8095. *Topics in Computer Systems* Advanced topics in computer systems, focusing on recent advances in areas such as advanced operating systems and networks, real-time operating systems, embedded systems, parallel and multicore systems, experimental computer systems, and system modeling and simulation. Credits: 3.

**Spring 2019:** *Approximate Methods to Compute Persistent Homology: Topological Data Analysis (TDA)* is the application of topological methods to analyze point cloud data. The fundamental idea is that topological methods can be used to study patterns or shapes that are preserved despite the presence of noise and variations in the data. The ability of TDA to identify shapes under certain deformations renders it immune to noise and leads to discovering properties of data that are not discernible by conventional methods of data analysis. *Persistent Homology* can be informally defined as a method for computing topological features of a data set at different spatial resolutions. It gives a multiscale view of the topology of a space by capturing the evolution of topology with increasing (or decreasing) resolutions. The computation of persistent homology is often the principal component in topological data analysis. Unfortunately the computation of persistent homology have exponential growth in both time and space. This challenge leads to the use of approximate methods to estimate the homology of large, high-dimensional point clouds. This class with study the general field of TDA and the various algorithms to compute and estimate the persistent homology of a point cloud.

**Instructor:** *P.A. Wilsey*
Office: 897 Rhodes
Phone: 556–4779

**Textbook:** Selected papers from the literature.

**Goals:** To expose students to Topological Data Analysis, Persistent Homology, and approximate methods for building solutions with applications on Big Data.

**Prerequisites:** Permission of instructor; advanced graduate standing; familiarity with abstract mathematics; exposure to data minining and data science; familiarity with computer systems and parallel/distributed computation; programming, data structures, and parallel algorithms.