CS 6170: Computational Topology
Topological Data Analysis for Data Scientists
Course Syllabus, Spring 2021

Instructor: Dr. Bei Wang Phillips
beiwang@sci.utah.edu

Course Information

Meeting Time: Tuesdays, Thursdays, 9:10am - 10:30am
Classroom: IVC via Zoom
Textbook: Computational Topology: An Introduction by Herbert Edelsbrunner and John Harer
Course Public Web Page:
`http://www.sci.utah.edu/~beiwang/teaching/cs6170-spring-2021.html`
Additional Materials for Students Available via Canvas.
Contact Information:

Instructor: Dr. Bei Wang Phillips
Office: WEB 4608
Email: beiwang@sci.utah.edu
Office Hours: See course webpage for details.

1 Course Description

Topological Data Analysis (TDA) is an emerging area in exploratory data analysis and data mining. It has had a growing interests and notable successes with an expanding research community. The application of topological techniques to large and complex data has opened up new opportunities in science, engineering and business intelligence. The goal of TDA is to understand complex datasets, where complexity arises from not only the massiveness of the data, but also from richness of the features. The objective of this class is to enable the students to become familiar with these new methods in TDA, from theory, algorithm and application perspectives.

Successful completion of the course will enable the students to become data practioners who can apply TDA pipelines to a variety of real-world datasets in material science, biomedicine, business intelligence, etc. The students can also pursue new research directions in the field of TDA and integrate advanced TDA techniques with other areas of data science such as data mining, machine learning, computer graphics, geometric modeling, mesh generation, and data visualization.

Suggested Topics: The course materials are organized under three mutually inclusive modules:

- TDA Foundations and Pipeline (FP)
- TDA, Machine Learning and Statistics (ML)
- TDA in Data Science (DS)

The course may cover (but is not limited to) the following topics:

- Basic concepts (graphs, connected components, topological space, manifold, point clouds)
- Combinatorial structures on point cloud data (simplicial complexes)
- New techniques in dimension reduction (circular coordinates, etc.)
- Clustering (topology-based data partition, classification)
- Homology and persistent homology
- Topological signatures for classification
- Structural inference and reconstruction from data
- Topological algorithms for massive data
- Multivariate and high-dimensional data analysis
- Topological data analysis for visualization (vector fields, topological structures)
- Practical applications of TDA

2 Prerequisites

There are no formal prerequisites for this class. Students will be expected to have basic knowledge of data structures and algorithmic techniques.

The targeted audience for the class includes PhD students, mater students and very-motivated upper level undergraduate students. The students are not required to be majoring in Computer Science, but it is preferable that the students have some background in algorithms and/or other data science related courses. (If you are not sure whether you are qualified to take this class, please email the instructor.)

3 Course Grading

- 2 assignments in the form of mini-projects (40 points, 40%; each project is worth 20 points). These projects are labeled as **Project 1 and 2** respectively in the course schedule.
- 1 final project (60 points, 60%). This is labeled as the **Final Project** in the course schedule.
  - Final project proposal (15 points, 15%)
  - Final project report (30 points, 30%), including the progress report (10 points) and the final report (20 points)
  - Final project presentation (15 points, 15%)
- Additional 10 bonus points may be available in the form of bonus assignment questions.

Scale for assigning letter grades is as follows (based on points). This scale might be curved based on overall class performance, while ensuring fairness to all.
A 100-93 **A-** 93-90
B+ 90-87 **B** 87-83 **B-** 83-80
C+ 80-77 **C** 77-73 **C-** 73-70
D+ 70-67 **D** 67-63 **D-** 63-60

E 60-0

Assignment Policies:

- Assignments are required to be done individually. Final projects can be done in groups. In general, discussing topics is allowed. However, the copying of each others' work is considered cheating and will result in a failing grade.
- There will be a call for assignments to be submitted (see course schedule for details). Assignments must be turned in at the beginning of class (i.e. at the time requested) on the day in which they are due.
- Most assignments should be submitted via Canvas. Each submission typically requires the following components (see each assignment description for details):
  - A PDF file (for project report)
  - A ZIP file for source code (if the assignment includes programming)
  - A URL that allows access to the deployed online software, or a URL that contains a link to a video that captures the software in action (these URLs are to be included in the submitted PDF file)
- Students are expected to submit completed assignments by the due date and time. To get full credit for an assignment, it must be turned in through Canvas by the start of class, specifically 9:10 am. Once the deadline is missed, those turned in late will lose 10% of its total points for each subsequent hour until it is turned in. Therefore, assignments will not be accepted more than 10 hours late, and will be given 0.
- For the portion of an assignment involving programming, if the programs do not execute, no partial points will be given. To demonstrate the execution of the program, a URL that allows access to the deployed online software, or a URL that contains a link to a video that captures the software in action should be provided.
- Please allocate sufficient time for completing the class assignments.
- For assignments (not including the final project), students can have a one-time two-day extension without penalty; please use this exception wisely.[1].
- For assignments, typesetting (Latex, MS Word, ... even a typewriter if you can find one) is required. Assignments deemed unreadable will be rejected at the time of collection; they can be resubmitted, but with the late penalty applied per the previously mentioned policy.

[1]Based on class discussion before Spring break, the one-time two-day extension is applicable for all assignments except for final project presentation and final project report

4 Policies and Guidelines

Please read carefully the School of Computing (SoC) policies and guidelines at: `http://www.cs.utah.edu/socguidelines/`. This document represents SoC policies and guidelines that the students should be aware of.

See also the College of Engineering Semester Guidelines at: `https://www.coe.utah.edu/wp-content/uploads/pdf/faculty/semester_guidelines.pdf`. These guidelines contain important dates regarding adding, dropping and withdrawing from classes as well as the College Policy regarding repeating courses.

Addressing Sexual Misconduct: Title IX makes it clear that violence and harassment based on sex and gender (which includes sexual orientation and gender identity/expression) is a civil rights offense subject to the same kinds of accountability and the same kinds of support applied to offenses against other protected categories such as race, national origin, color, religion, age, status as a person with a disability, veteran's status or genetic in-formation. If you or someone you know has been harassed or assaulted, you are encouraged to come speak to the School of Computing Advisors and/or to the Title IX Coordinator in the Office of Equal Opportunity and Affirmative Action, 135 Park Building, 801-581-8365, or the Office of the Dean of Students, 270 Union Building, 801-581-7066. For support and confidential consultation, contact the Center for Student Wellness, 426 SSB, 801-581-7776. To report to the police, contact the Department of Public Safety, 801-585-2677(COPS). More information is available at `https://safeu.utah.edu`.

5 Final Project

5.1 Project Description

Your final project can be designed from the perspective of a data practitioner, a developer or a data theorist. As a data practitioner, you could use interesting and nontrivial datasets in various application domains (e.g. marketing, scientific simulation, transportation, business intelligence, etc.), and apply emerging and innovative TDA techniques (possible in combination with other data mining and machine learning techniques) to obtain insights on the data. As a developer, you could develop new software tools or extend existing ones that combine TDA with data mining and machine learning. As a data theorist, you could work towards new theories, algorithms or data structures in the field of computational topology and TDA.

You are responsible to pitch your project idea at a level that is appropriate for your background. Try to challenge yourself, at the same time, be realistic. In the case when you underestimate the difficulty of your project, please make sure that you have something to submit by the due date; choosing a project that is too difficult is not a valid reason for an incomplete.

5.2 Project Team

You will work in a team with two members (forming a team with one member, or a team with more than two members will require the permission from the instructor). You can form the team on your own; or the instructor could make some suggestions based on your submitted CV. Please try to form your project team as early as possible.

5.3 Important Dates

There are several milestones for the final project.

- Project team creation, typically in February, 2021.
- Project proposal description, typically in March, 2021.
- Project progress report, typically in March/April, 2021.
- Project presentations, typically in April/May, 2021.
- Project final report, typically in April/May, 2021.

See Canvas for details on the due dates.