CS 6965: Advanced Data Visualization

The Study of Large and Complex Data

Course Syllabus, Spring 2018

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

Course Information

Meeting Time: Tuesdays, Thursdays, 9:10am - 10:30am
Classroom: WEB L120
Textbook: None
Web page: `http://www.sci.utah.edu/~beiwang/teaching/cs6965-spring-2018.html`

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

1 Course Description

Data visualization is an integral part of data analysis; think about wine and cheese, they just go hand in hand. In this course, we would discover how new and advanced data visualization tools offer analytics capabilities that can help us understand large and complex data. Large and complex data arise from networks, high-dimensional point clouds, multivariate functions, heterogeneous personal data and ensembles; as such, this course is very much data-driven, as our topics are divided into modules which focus on particular data modality. The objective of this class is to enable the students to become familiar with innovative techniques that combine data analysis with data visualization, from algorithmic and implementation perspectives.

This course in going to focus on using existing libraries and developing new tools to study large complex data, in particular, network data such as social and biological networks, heterogeneous personal data, high-dimensional and multivariate data that arise from science, social studies and business intelligence. The course is heavily project based, with multiple mini-projects and a collaborative final project.

Successful completion of the course will enable the students to pursue new research directions in data analysis and data visualization; and apply emerging and innovative techniques to data in various application domains.

Suggested Topics: The course will cover (but is not limited to) the following topics:

- High-dimensional data: machine learning and visualization.

- Visualizing large graphs and networks.
- Topological abstraction and summarization for data visualization.
- Personalized visualization: humanistic approach to data.

2 Prerequisites

There are no formal prerequisites for this class. Students, however, will be expected to have basic knowledge of data structures and algorithmic techniques, bachelor-level knowledge in mathematics or computer science, and working knowledge of programming, ideally with Python and/or C++.

The targeted audience for the class includes PhD students, master 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, and have working knowledge of programming, ideally with Python and/or C++. If you are not sure whether you are qualified to take this class, please email/talk to the instructor.

3 Course Grading

- 4 assignments in the form of mini-projects (60 points, 60%; each project is worth 15 points). These projects are labeled as **Project 1, 2, 3 and 4** respectively in the course schedule.
- 1 final project (40 points, 40%). This is labeled as the **Final Project** in the course schedule.
  - Project proposal (10 points, 10%)
  - Project report (25 points, 25%), including the progress report (5 points) and the final report (20 points)
  - Project presentation (5 points, 5%)
- 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.

- 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.

4 Policies and Guidelines

Please read carefully the School of Computing (SoC) policies and guidelines at:
`https://www.cs.utah.edu/~germain/SoC_Guidelines_Spring_2017`.
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.

5 Final Project

5.1 Project Description

Your final project can be designed from the perspective of an expert user or from the perspective of a developer. As an expert user, 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 techniques. As a developer, you could develop new software tools or extend existing ones that combine machine learning with visualization. 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

You could submit reports for the following project milestones on or before (as early as possible) the due dates. Each milestone report is due at 9:10 a.m. on its due date; the usual late submission policy applies.

- Project team creation: due February 8.
- Project proposal description: due March 6.
- Project progress report: due March 27.
- Project final report: due April 24.
- Project presentations: on April 24 (9:10 - 10:30 a.m.) and April 27th (8:00 - 10:00 a.m.)

You should form your two person project team by Feburary 8.

Your project proposal description is due on March 6 to give sufficient time for the instructor feedback. Your proposal should be 3-4 pages, well thought out and well-written (see course website for details).

Your project progress report is due on March 27. The report should describe the portion of the project that has been completed, what yet to be done. It should describe whether or not your team has met the milestones described in the proposal. If you plan to make modifications to your project proposal, please provide a reasonable justification and an updated timeline with milestones (see course website for details).

The final project report is due on April 24 (see course website for details).