CS 6959 Computational Inverse Problems
Spring Semester 2014

Instructor: Chris Johnson, Ph.D.

Time: T,Th 12:25 - 1:45 p.m.

Place: WEB L110

Office: 3850 WEB

Phone: 801-581-7705

Email: crj@sci.utah.edu

Class web page: www.eng.utah.edu/~cs6959

Co-Instructor: Yaniv Gur, Ph.D.

Email: yanivg@sci.utah.edu

Text: *Discrete Inverse Problems: Insight and Algorithms* by Per Christian Hansen.

Course Philosophy: The study of inverse problems is very new – and very old. The latest medical imaging devices are essentially inverse problem solvers; they reconstruct two- or three-dimensional objects from projections. More than two thousand years ago, in book VII of his *Republic*, Plato posed essentially the same problem in his allegory of the cave, namely, he considered the philisophical implications of reconstructing "reality" from observations of shadows cast upon a wall.

Inverse problems can be found in virtually all engineering and scientific subdisciplines from applications in geophysics to mechanical engineering to medicine.

We will take a computational approach to studying inverse problems concentrating on algorithms focusing on reliability and efficiency, both of which are important as the size and complexity of the computational problems grow.

Course Goals: Upon completion of this course, the student should

- Know commonly used algorithms and techniques for analyzing and computing inverse problems,
- Have seen and discussed examples of inverse problems in a variety of fields,
- Know where to locate further inverse problem resources and references,
- Have completed a computational inverse problems project.

Assignments: There are two main types of assingments for this course. One is in the form of in between class homework that will primarily consist of using the Matlab Regularization Toolkit.

The second will be in the form of a computational inverse problems project. This project can involve data you have collected from a simulation and/or experiment, development of a new software tool(s) for inverse problems, or a theoretical investigation of a topic in inverse problem theory. My goal is to supply the student with as close to *real life* inverse problem research applications as possible within the confines of a semester long class.

Languages: For this course we will primarily use Matlab along with the Regularization Toolkit.

Grades: Final course grades will be computed according to 80% Homework and Labs and 20% Final Project.

Incompletes: As the project is due by the end of the semester, in past similar project-based courses, it has turned out that some people do not wisely schedule their time and do not finish their projects. They then want to take an incomplete and finish the project sometime in the summer. I only give incompletes very rarely and only for truely unusual circumstances (death in the family, etc.), so **please** work to finish your final project on time.

Syllabus for CS 6959

See www.eng.utah.edu/~cs6959/schedule.html for an updated schedule.

Week 1. Overview of inverse problems in science, engineering, and medicine.

Week 2. Discrete inverse problems

Week 3. Regularization methods

Week 4. Regularization parameters

Week 5. Solving real inverse problems

Week 6. Iterative regularization methods

Week 7. Guest lectures

Week 8. Image deblurring

Week 9. Spring break

Week 10. Large-scale inverse problems

Week 11. Statistical inverse problems

Week 12. Case studies

Week 13. Project presentations

Week 14. Guest lectures

Week 15. Final project due

Note: During the semester, we will have several guest lectures on selected topics in inverse problems.

Computational Inverse Problem Project

Due dates: Project description **due March 6**. Project presentations will be on April 15 and April 17. The final project write up is **due April 30**.

The computational inverse project can be (1) from simulation and/or experimental data you have, (2) development of new software tools for inverse problems and/or creation of a new module(s) for an exisiting inverse package (such as the Regularization Toolkit), or (3) conducting a theoretical investigation of some aspect of computational inverse problems and writing a paper discussing your results.

It is your responsibility to pitch your project at the appropriate level. Challenge, but do not exhaust, yourself. Please ensure that even if you underestimate the difficulty of your project, you will have something to hand in by the due date (choosing too difficult a project is not a valid reason for an incomplete).

Group projects are allowed, however, the size and difficulty of the project should reflect the number of people involved in a single project.

On March 6 your project design report is due. This should be a well thought out, well-written one page description of your proposed project. It should outline any necessary background, specifically what goals you plan on accomplishing, and what you will need to do in order to accomplish your goals. You will also need to include what software/hardware you plan to use, and what you intend to hand in (i.e. what are the “deliverables”).

In grading the projects, I will be looking for a well-designed, substantial, interesting project. Furthermore, your implementation, content and style of the final results should be of high quality. A final criteria for grading is that the progress report and final report are handed in on time.

You will present your final projects on Tuesday, **April 15** and Thursday, **April 17**. A final project sign-up sheet will be handed out in class for you to schedule a time. Presentations should typically take approximately 20 minutes.

Project Design Report

Please hand in your Project Design Report by **March 6** (or sooner). It should contain the following information.

Student Name(s):

Project title:

- Give an overview of the project.
- Why is this project important and/or interesting?
- If you are doing a programming project, list the hardware and software you will be using:
- What have you done thus far and what will you have to do to complete this project?
- When the project is completed, how *specifically* can we evaluate how successful it is?
- Any other useful information:

Project Final Report

You will be required to hand in your Project Final Report on **April 30**. Your final report should contain the following information.

Student Name(s):
Project title:

- Provide a brief description of your project and how to run it if it is not self-explanatory.
- Outline what you learned from doing this project.
- If you have not accomplished all the goals of your project, or if you have exceeded them, describe how the finished project differs form the description in your project design.
- Evaluate your project: how successful do you think it was? What are the strengths and weaknesses of your project?
- Any other comments useful for me in evaluating your project: