CS 6962 Computational Inverse Problems
Fall Semester 2024

Instructor: Chris Johnson, Ph.D.

Time: Mondays and Wednesdays between 1:25 - 2:45 p.m.

Place: WEB 2760

Office: WEB 4692

Phone: 801-587-7875

Email: crj@sci.utah.edu

Class web page: `https://my.eng.utah.edu/~cs6962`

Teaching Assistant: TBA

Email: TBA

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

Course Description: Inverse problems can be found in virtually all engineering and scientific sub-disciplines from applications in geophysics to mechanical engineering to medicine. The solution of an inverse problem entails determining unknown causes based on observation of their effects. The latest medical imaging devices are essentially inverse problem solvers; they reconstruct two- or three-dimensional objects from projections. After an introduction and overview of inverse problems, we will concentrate on computational algorithms for solving inverse problems.

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 next semester. 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 6962

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. Fall break

Week 9. Image deblurring

Week 10. Large-scale inverse problems

Week 11. Statistical inverse problems

Week 12. Quantifying uncertainty in inverse problems

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 October 4.** Project presentations will be on December 2 and December 4. The final project write up is **due December 11**.

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, **December 6** and Thursday, **December 8**. 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 **October 4** (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?
- What are the objectives of the project? What are the questions you want to answer?
- What would you like to learn by completing this project?
- What data will you be using for your project?
- If you are doing a programming project, list the hardware and software you will be using.
- What is your project schedule? What have you done thus far and what will you have to do to complete this project? Be as specific as possible.
- 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 **December 11**. Your final report should contain the following information.

Student Name(s):
Project title:

- Overview and goals of your project.
- Background and related work. What books, papers and websites did you learn from?
- Provide a description of your project. What data did you use? What questions did you answer? Describe any new questions that arose throughout the project.
- Discuss the implementation details of your project.
- 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?
- Provide additional comments useful in evaluating your project.

References

- Discrete Inverse Problems: Insight and Algorithms, P.C. Hansen, SIAM Press, 2010. The main book for the course.
- An Introduction to Inverse Problems with Applications, F.D.M. Neto and A. J. Neto, Springer, 2013. (A nice introduction to inverse theory from a mathematical point of view.).
- Rank-Deficient and Discrete Ill-Posed Problems, P.C. Hansen, SIAM Press, 1998. (This is an excellent book on discrete linear inverse problems).
- Linear and Nonlinear Inverse Problems with Practical Applications, Jennifer L. Müller and, Samuli Siltanen, SIAM Press, 2012. (This is very nice introductory book with several good applications).
- Parameter Estimation and Inverse Problems, Second Edition, Richard C. Aster, Brian Borchers, and Clifford H. Thurber, Academic Press, 2012. (This is very nice introduction to inverse problems with a geophysics bent).
- Inverse Problem Theory and Methods for Model Parameter Estimation Albert Tarantola, Siam Press, 2004. (This is a classic text on probabilistic inverse theory).
- Geophysical Inverse Theory and Regularization Problems, Michael S. Zhdanov, Elsevier, 2002. (This is a tour de force on geophysical inverse problems written by a Utah Geophysics Professor).
- Large-scale Inverse Problems and Quantification of Uncertainty, L Biegler, G. Biros, O. Ghattas et al. editors, Wiley, 2011. (This is a nice collection of papers on Bayesian techniques with a few large-scale applications.).
- Optimization and Regularization for Computational Inverse Problems and Applications, Y. Wang, A.G. Yagola, and C. Yang, editors, Springer, 2010. (A collection of papers that treat inverse problems as optimization problems).
- Inverse Problems in the Mathematical Sciences, C.W. Groetsch, Vieweg Mathematics for Scientists and Engineers, 1993. (This is a really nice introduction to inverse problems and has an excellent annotated bibliography).
- Statistical and Computational Inverse Problems, J. Kaipio and E. Somersalo, Springer, 2005. (A nice introduction to statistical inverse theory).
- Computational Uncertainty Quantification for Inverse Problems, Johnathan Bardsley, 2018. (A good introduction to uncertainty techniques in inverse problems).
- Geophysical Data Analysis: Discrete Inverse Theory, W. Menke, Academic Press, 3rd edition, 2012 (A nice introduction to inverse theory within the context of geophysical problems).

- Computational Inverse Problems in Electrocardiography, Peter R. Johnston, WIT Press, 2001. (A nice collection of articles in inverse cardiology).
- Mathematically Modeling the Electrical Activity of the Heart: From Cell to Body Surface and Back Andrew J. Pullan, World Scientific Publishing Company, 2005. (A nice book on computational forward and inverse problems in cardiology).
- Handbook of Neural Activity Measurement, R. Brette and A. Destexhe, editors, Cambridge University Press, 2012. (Chapter 6 of this book is on MEG and EEG source estimation).
- Numerical Methods for the Solution of Ill-Posed Problems, A.N. Tikhonov, A.V. Goncharsky, V.V. Stepanov, and A.G. Yagola, Kluwer Academic Publishers, 1995 (a reprint of a 1990 Russian version). (This is a classic by one of the founders, Tikhonov, of regularization theory. It contains lots of Fortran code. Note that it is very expensive).
- An Introduction to the Mathematical Theory of Inverse Problems, A. Kirsch, Springer, 1996. (A nice overview of the mathematical analysis of classical inverse problems).
- Ill-Posed Problems: Theory and Applications, A. Bakushinsky and A. Goncharsky, Kluwer Academic Press, 1994 (a reprint of a 1989 Russian version). (A collection of chapters that use the concept of the “regularizing algorithm").
- Inverse and Ill-Posed Problems, Heinz Engl and C.W. Groetsch, editors, Academic Press, 1987. (An edited collection of papers on several aspects of inverse problems including theory and applications. Thisis an often cited collection).
- Conjugate Gradient Type Methods for Ill-Posed Problems, Martin Hanke, Pitman Research Notes in Mathematics Series 327, Longman Scientific and Technical, 1995. (This is a short monograph that contains recent efforts in iterative Krylov subspace type methods for inverse problems).
- Ill-Posed Problems in the Natural Sciences, A.N. Tikhonov and A.V. Goncharsky, editors, MIR Publishers, 1987. (This is a nice collection of inverse application papers).
- Regularization Methods for Ill-Posed Problems, V.A. Morozov, CRC Press, 1993. (This book is a mathematical treatise on regularization).
- Inverse Problems in Partial Differential Equations, D. Colton, R. Ewing, and W. Rundell, editors, 1990. (A collection of application papers and applied mathematics papers on inverse problems in various pdes).
- Plato's Cave and Inverse Problems, www.mlahanas.de/Greeks/PlatosCave.htm

MATLAB

- http://www2.imm.dtu.dk/~pcha/Regutools/Regularization Toolkit for Matlab
- http://www.mathworks.com/moler/chapters.htmlNumerical Computing with Matlab This book is a very nice overview of numerical analysis with several examples using Matlab. The book is available for free on-line. The Matlab codes used in the book are also available on-line.
- MATLAB Guide by Desmond J. Higham and Nicholas J. Higham, SIAM Press, 2005.
- Mastering MATLAB by Duane C. Hanselman and Bruce L. Littlefield, Prentice Hall, 2011.
- http://www.gnu.org/software/octaveOctave - an open source, freely available alternative to Matlab
- Look at the Plot Catalog in MATLAB to view the various types of plots, or see the online documentation here:
  http://www.mathworks.com/access/helpdesk/help/techdoc/creating_plots/f9-53405.html
- Tutorials from MathWorks (developer of MATLAB), with links to other tutorials:
  http://www.mathworks.com/academia/student_center/tutorials/launchpad.html
- Users Guide (command, language, and object reference):
  http://www.mathworks.com/access/helpdesk/help/techdoc/matlab_product_page.html
- Good tutorial on the basics of matrix and vector operations and programming (very little on plotting):
  http://amath.colorado.edu/computing/Matlab/Tutorial/
- Tutorial focused on plotting basics:
  http://www.engin.umich.edu/group/ctm/extras/plot.html

SCI Institute Software

- http://software.sci.utah.edu/SCIRun Software System: A scientific problem solving environment for modeling, simulation and visualization developed by the Scientific Computing and Imaging Institute at the University of Utah.
- http://www.sci.utah.edu/devbuilds/scirun_docs/ECGToolkitGuide.pdfSCIRun Forward/Inverse ECG Toolkit: This toolkit is a col- lection of modules and networks within the SCIRun system, which can be used to solve forward and inverse electrocardiography problems.

Python

- Python Scripting for Computational Science by Hans Petter Langtangen, Springer, 2004.
- Python Essential Reference (3rd Edition) by David M. Beazley, Sams, 2006.
- Enthought Python Distribution (EPD), which includes the NumPy, SciPy, matplotlib, mlab, Mayavi2, and other libraries, plus useful tools such as the IPython interpreter shell (with features such as code completion). Free for academic use.
  `http://www.enthought.com/products/epd.php`
- SciPy Manual:
  `https://docs.scipy.org/doc/scipy/reference/`http://www.scipy.org/SciPy API
- NumPy Documentation:
  `https://numpy.org/doc/`http://www.scipy.org/NumPy Documentation
- matplotlib tutorial (2D plots):
  `http://matplotlib.sourceforge.net/users/pyplot_tutorial.html`
- mlab documentation (3D plots):
  `http://code.enthought.com/projects/mayavi/docs/development/html/mayavi/mlab.html`
- Python documentation:
  `http://www.python.org/doc/`

Introductory Numerical Analysis

- Scientific Computing: An Introductory Survey, Second Edition by Michael T. Heath, published by McGraw-Hill, New York, 2002.
- Scientific Computing with MATLAB by Alfio Quarteroni and Fausto Saleri, Springer, 2003.
- Online numerical analysis text covering a variety of topics:
  `https://ece.uwaterloo.ca/~dwharder/NumericalAnalysis/Contents/`
- Another set of numerical analysis notes with a wealth of other resources:
  `http://numericalmethods.eng.usf.edu/`

Statistics

- Data Analysis: A Bayesian Tutorial, Devinderjit Sivia and John Skilling, Oxford University Press, 2006. (This is a nice introduction to Bayesian data analysis).
- Demo of computing π by a Monte Carlo approximation:
  `http://polymer.bu.edu/java/java/montepi/`

- Another set of overview slides for Monte Carlo integration, with sample code in C (skip to page 5 for the relevant part):
  http://www.sph.umich.edu/csg/abecasis/class/2006/615.22.pdf