Fall 2020
Numerical Optimization MAD 5420

Details

- Time and Place: MWF 12:20–13:10, on-line Zoom lectures
- Instructor: K. A. Gallivan (5-0306, 318 Love Building, gallivan@math.fsu.edu)
  - Homepage: www.math.fsu.edu/~gallivan
  - Office Hours: by appointment (arrange time via email)
- Prerequisites: MAC2313 (Calculus 3), MAP3105 (Linear Algebra), MAD3703 (Numerical Analysis 1), programming proficiency.
- No textbook is required however there are recommended reference texts that are useful study resources.
- Reference Texts:
  2. Introduction to Nonlinear Optimization, A. Beck, SIAM
  3. Iterative Methods for Optimization, C. T. Kelley, SIAM
  4. Optimization by Vector Space Methods, D. G. Luenberger, Wiley
  8. Primal-Dual Interior-Point Methods, S. J. Wright, SIAM
- Class Information: Class notes, homework, programming assignments and announcements will be posted on the class website (follow the teaching link from www.math.fsu.edu/~gallivan). You are expected to consult the website in a timely and regular manner. The user and password information for portions of the class website will be sent to those registered in an email/Canvas announcement. If you do not receive this information before the first class contact the instructor. Canvas will be used to post zoom information about the lectures, to join the zoom lectures and for class email announcements.
- Other books, papers, and software resources will be cited, posted or linked in the class notes, the homework, and on the class website. These are background material and suggested references. You will be expected to read a small subset identified in the lectures. The others are for your graduate academic development.
• Lectures will be given on zoom. The meeting information will be posted weekly on the class Canvas page. Initial zoom information will be sent in email/Canvas announcement to those registered.

• Students outside the USA in time zones that are not conducive to attending the real-time zoom lectures at 12:20 Eastern Time should contact the instructor and alternative arrangements will be made.

• Class Participation Policy: Participation in class by asking questions and responding to questions posed by the instructor is expected and strongly encouraged. The diverse background of the students implies several students will be unfamiliar with each topic and clarification by such questions and responses is vital to understanding. Students are expected to be prepared for class lectures and for any office visits. Students are encouraged to prepare carefully for class by reading relevant posted notes and sections of the textbook.

• Meetings with the instructor are expected and encouraged especially for discussions that are too lengthy for class. It is particularly important to have such meetings early and often for material about which you are uncertain. These will be held on zoom at agreed upon times. Group meetings are encouraged.

• Grades: Homework (programming and written problems) 100%

• There are no exams.

• Electronic devices may be used to access class notes and related material during lectures. Other uses of cell phones and similar devices are often disruptive to the lecture and are not permitted.

• Homework: Homework will be assigned regularly consist and will comprise written exercises and programming assignments.

• Solutions to the programming assignments are preferred to be in a compiled and typed language such Fortran, C, C++, Java. Julia is also acceptable but variable types and data structures are expected to be defined and constructed explicitly. However, Julia, Python and environment-based programs in, e.g., MATLAB, are also acceptable.

• Plagiarism is a violation of the university honor code. With respect to the solutions of programming problems in the homework, it is not acceptable to engage in plagiarism. You may discuss the programming problems with each other but any significant discussion must cited, i.e., it should be treated like the citation of any outside material used in your solutions. Such a citation must include names and the substance of the discussion. All students must design and implement their own code. All students must write individually the description of the code, its complexity, the experimental design, the empirical results and the interpretation of the results. No student should provide any portion of their code to any other student in class. If you find a library code (not written by a student in class) that performs a portion of the task it must be cited — including the specific source of the code and its function. You are still responsible for describing correctly its implementation and time/space complexity in your solutions. Credit for the program will be prorated based on the amount of functionality performed by a cited
library code relative to the functionality required to solve the assigned problem. **Citations when writing solutions to the analytical problems included in a graded homework or programming assignment and applicable penalties for plagiarism are covered by these same policies.** The first offense of submitting a solution without the appropriate citation will result in 0 credit for the programming/experimental portion of the assignment or the particular solution to the analytical problem. Multiple offenses may result in referral to the university for discipline according to university regulations.

• University Attendance Policy: Excused absences include documented illness, deaths in the family and other documented crises, call to active military duty or jury duty, religious holy days, and official University activities. These absences will be accommodated in a way that does not arbitrarily penalize students who have a valid excuse. Consideration will also be given to students whose dependent children experience serious illness.

• Class Attendance Policy: With the exception of the first class meeting, attendance is not required but it is strongly advised. A student absent from class bears the full responsibility for all subject matter and procedural information discussed in class.

**Content**

This course is an introduction to the theory and algorithms to solve continuous optimization problem. The topics include, as time permits:

1. Hilbert space theory and basic algorithms for approximation problems.
   - Approximation, optimality, projection, and orthogonality.
   - Parameterization for an unconstrained optimization form.
   - The Normal Equations.
   - Transformation-based structure modification approach.
   - Incremental observation updates.
   - Relationship to unconstrained convex quadratic problems.

2. Unconstrained optimization on $\mathbb{R}^n$
   - Optimality conditions on $\mathbb{R}^n$.
   - Line search algorithms on $\mathbb{R}^n$.
     - Descent and conjugate direction algorithms for convex quadratic problems on $\mathbb{R}^n$.
     - Descent and termination criteria.
     - Newton’s method and its efficient variants.
     - Quasi-Newton algorithms.
     - Limited memory Quasi-Newton algorithms.
     - Nonlinear conjugate gradient algorithms.
   - Introduction to trust region algorithms on $\mathbb{R}^n$.

3. Linear Programming
• Convexity and optimality conditions for linear programming on $\mathbb{R}^n$
• Simplex algorithm
• Interior point and penalty methods introduction

4. Constrained optimization on $\mathbb{R}^n$
• Optimality conditions for optimization over a convex feasible set in $\mathbb{R}^n$.
• Gradient projection for a convex feasible set on $\mathbb{R}^n$.
• KKT Optimality conditions for nonlinear programming on $\mathbb{R}^n$.
• Constraint qualifications.
• Basic nonlinear programming algorithms on $\mathbb{R}^n$.

5. Proximal gradient methods.


7. Application examples.

Objectives

The material covered in the class covers tools that have been developed to solve problems numerically. Thus, you will be expected to be able to analyze, design, and implement some of the techniques and to solve problems with them. In addition, you will be expected to know how the methods were derived and when they are and are not appropriate to use.

Syllabus Changes

Except for changes that substantially affect implementation of the evaluation (grading) statement, this syllabus is a guide for the course and is subject to change with advance notice.

Honor Code

The Florida State University Academic Honor Policy outlines the University's expectations for the integrity of students academic work, the procedures for resolving alleged violations of those expectations, and the rights and responsibilities of students and faculty members throughout the process. Students are responsible for reading the Academic Honor Policy and for living up to their pledge to ... be honest and truthful and ... [to] strive for personal and institutional integrity at Florida State University. (Florida State University Academic Honor Policy, found at https://dof.fsu.edu/honorpolicy.htm.)

Americans with Disabilities Act

Students with disabilities needing academic accommodation should during the first week of class:

1. register with and provide documentation to the Student Disability Resource Center;
2. contact the instructor indicating the need for accommodation and what type.

This syllabus and other class materials are available in an alternative format upon request. For more information about services available to FSU students with disabilities, contact the:

Student Disability Resource Center
874 Traditions Way
108 Student Services Building
Tallahassee FL, 32306-4167
644-9566 (voice), 644-8504 (TDD), sdrc@admin.fsu.edu, http://www.disabilitycenter.fsu.edu.