AMSC 607/ CMSC 764 TTh 11-12:15, CSI 3120  
Advanced Numerical Optimization, Fall, 2006  
http://www.cs.umd.edu/~oleary/a607

Prof. Dianne P. O'Leary: Room 3271 A.V. Williams Building, (301) 405-2678, oleary@cs.umd.edu http://www.cs.umd.edu/~oleary/

Office Hours: Tuesday 8:45-9:15, Thursday 8:00-9:15, Friday 9-10, and by appointment, in AVW 3271.  
Please restrict telephone inquiries to office hour times, except in “emergencies.” E-mail is welcome anytime!

Prerequisites: A course in linear algebra and a course in numerical analysis.  
(A previous course in optimization is NOT expected.)

Texts (available on-line):


Grading: Grading will be on a *curve*, except that you will be guaranteed an A if your average is 90% or better, a B if your average is 80% or better, etc. Keep all of your work in case there is any question about recording of grades.

<table>
<thead>
<tr>
<th>Component</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework</td>
<td>approx. 200</td>
</tr>
<tr>
<td>Project</td>
<td>100</td>
</tr>
<tr>
<td>Final Exam</td>
<td>None</td>
</tr>
</tbody>
</table>

Homework: 1-2 weeks will be allowed for each assignment. Some homework will include programming assignments using the software package MATLAB. To pass this course, you must make an honest effort at each homework. Partial credit will be given for partially-working programs. There will be a 15% penalty for assignments turned in up to 2 days late, 30% penalty for assignments turned in 2-4 days late, etc.
The project will involve writing a review of an article and possibly experimenting with an algorithm. More information will be given in early November. It is due at noon on Thursday, December 14.

Regrades: If you think a mistake has been made in grading your work, submit it for regrading within two weeks of the date on which the work was returned to the class. After that, the grade will be considered final.

News: Assignments, course notes, answers to homeworks, and announcements will be posted on the course’s homepage. You are responsible for checking this site before each class.

Academic Integrity: Class accounts are to be used only for class assignments. All files within the accounts are subject to inspection, and the campus code of computer conduct must be followed. All work that you submit in this course must be your own; group efforts will be considered academic dishonesty. See http://www.studenthonorcouncil.umd.edu/code.html for definitions and sanctions. You may discuss homework and the project in a general way, but you may not consult any one else’s written work, program drafts, computer files, etc. Any marked similarity in form or notation between submissions with different authors will be regarded as evidence of academic dishonesty – so protect your work. You are free to use reference material to help you with assignments, but you must cite any reference you use. Such citation will not lower your grade, although extensive quotation might.
COURSE OUTLINE

Objective: Understand and use state-of-the-art algorithms for optimization. (Note that this is a moving target.)

Prerequisites: A course in linear algebra and a course in numerical analysis. A previous course in optimization is not assumed.

1. Introduction (Nash & Sofer Chapters 1-3) (approx. 2 weeks)
   - sources of optimization problems
   - feasibility, optimality, convexity
   - representation of linear constraints

2. Unconstrained Optimization (Nash & Sofer Chapters 10-12) (approx. 4 weeks)
   - A survey of the characterization of solutions to minimization problems, algorithms for problems of various smoothness, and global optimization.
   - optimality conditions, descent methods
   - Newton’s method and steepest descent
   - automating differentiation
   - variations on Newton’s method

3. Constrained Optimization (Nash & Sofer Chapters 14-17, Nemirovski notes) (approx. 8 weeks)
   - A survey of feasible set methods, projection methods, and interior point methods for linear and nonlinear constraints.
   - optimality conditions
   - duality
   - feasible point methods
   - penalty and barrier methods
   - interior point methods