Projects are due in electronic form by 11:59pm, Friday December 14, 2012

Purpose of the project. Students are to complete and submit a written report about an individual “mini-research” project involving a topic relevant to numerical optimization. Each project report should demonstrate your engagement with a topic not covered in class in detail, and should discuss what you learned during the project, explicitly pointing out your original contributions. All projects must involve computing (both numbers and associated explanations) based on your code and your numerical experiments.

Any project of which major portions have been submitted in one or more previous classes cannot be re-used for the project in this class.

Required advance approval. By 11:59pm on Friday, November 16, 2012, every student must have received individual approval of his/her project from me via an email message. Before approving a project topic, I will need to receive an email message from you in which you describe your project in general terms. If your project fits into one of the categories listed below, please specify that topic in your description; if not, please say that it does not fit into one of the broad topics listed below. No credit will be given for projects on topics that I have not explicitly approved in an email message to you.

Form of the project. A project should have the ingredients of a scientific paper (title, your name and affiliation, abstract, project body, and bibliography). The project body will typically consist of approximately 7–10 pages that describe the topic and summarize your investigations, including theoretical analysis and computational experiments. A supplement should be provided containing your code (or code from others, properly attributed) that was used in the project.

Two excellent projects from previous courses (Numerical Optimization and Numerical Computing) will be posted on the course website to provide guidance about form and the hoped-for level of quality.

The project’s role in your course grade. For students who took the midterm, as discussed in class, there are three weights (25%, 35%, 40%), which will be associated with your three scores in (homework, midterm, project) so that your score is maximized. Your course grade will be based on the weighted sum.

For students who missed the midterm because of illness, there are two weights (60%, 40%), which will be associated with your scores in two areas (homework, project) so that your score is maximized. Your course grade will be based on the weighted sum.

Evaluation criteria. Projects will be graded on (1) knowledge, understanding, and creativity related to your topic, (2) clarity and correctness of explanations and examples, (3) insightfulness of your discussion and (4) originality. Your grade will be based on both scientific content and quality of exposition.

Citing the work of others. I expect everyone to seek references and information, from and beyond the initial pointers mentioned below. In this context, it is essential to include explicit citations to any and all reference material used in the project, including material from the Web. If you fail to cite work by someone else that you use in your project, this will be considered plagiarism and your score on the project will be zero.
The following list provides ideas for possible projects in various areas, but it is not even close to complete. If you wish to do your project on some other topic, that is fine as long as it meets the criteria described above and I have approved the project in advance.

Each numbered item on the list includes enough material for more than one project. If different students want to work on one specific topic, we will need to decide in advance, and confirm via email, how the content will be divided among you.

1. **Optimization in finance.**

   This topic is so huge that it’s not possible to include an adequate list of references here. Information about possible topics can be gleaned from Google hits, from looking at recent papers in the SIAM Journal on Financial Mathematics (epubs.siam.org/journal/sjfmbj), and from

   www.math.nyu.edu/financial_mathematics/content/02_financial/Working_Paper_Series.html
   www.watrisq.uwaterloo.ca/

   Matlab has a Financial Toolbox, which can be reached from your Courant account by using the “help” tab and then typing “financial toolbox”. One of the sections of possible interest is on “portfolio optimization”.

   There are many other websites, but not all are reliable. You could even consider a project on the poor advice given about numerical issues in some of these websites, papers, and books.

2. **Modern implementations of the simplex method.**


   or.journal.informs.org/content/50/1/3.abstract

   This paper discusses the techniques used in CPLEX as of 2002 to make the simplex method faster. Bixby has since started Gurobi, a company that provides optimization software for a variety of problems, including linear programming. A project on modern LP software could examine one of the techniques discussed in www.gurobi.com/resources/getting-started/lp-basics, or

   - steepest-edge pricing;
   - modern linear algebra in simplex and interior methods; see the references listed on the website for a class taught at Stanford by Michael Saunders:
     www.stanford.edu/class/msande318/references.html
   - characteristics of LPs for which simplex methods are generally more effective, and problems for which interior methods are generally more effective;
   - how LPs are used as subproblems within integer or mixed-integer programming problems.

   You are not expected to solve LPs with millions of variables, but even for moderate-size problems interesting experiments are possible; see David Eigen’s 2011 project, posted on the current course website.

3. **Treatment of an indefinite Hessian.**

   In a Newton-based method for unconstrained optimization, there is no “right answer” for what should be done when the Hessian is indefinite, or even when it is positive semidefinite. The 2008 paper “Modified Cholesky algorithms: a catalog with new approaches”, by Haw-ren Fang and Dianne O’Leary, Mathematical Programming A 115, 319–349, www.springerlink.com/content/q4m5438141j0v028, discusses some of the major linear algebraic issues.
As an alternative, many optimization experts believe that problems with an indefinite Hessian are best solved using a trust region method. An excellent overview of modified Newton and trust region methods is given in lecture notes by Mihai Anitescu of Argonne National Laboratory, which can be accessed by Googling modified Hessian trust region (look at “Lecture 4”) and modified Hessian trust region anitescu (look at “Lecture 4.1 Fundamentals”).

A project in this area should include consideration not just of the linear algebra issues, but also of how well different techniques work in practice on optimization problems with indefinite Hessian matrices.

4. Derivative-free optimization.

In recent years there has been great interest in optimization methods that do not require any derivatives from the problem functions. The book Introduction to Derivative-Free Optimization, by Conn, Scheinberg, and Vicente, SIAM, 2009, provides a good overview of these methods, and numerous talks on this topic were given at the recent International Symposium on Mathematical Programming (ISMP) in Berlin, August 2012 (ismp2012.mathopt.org/). Looking at the titles of the talks given in the track on “Derivative-free and simulation-based optimization” should suggest some possible topics.

Jorge Moré and Stefan Wild at Argonne National Laboratory (Argonne, Illinois) have an interesting website about what they call a “shootout” (a comparison) of derivative-free optimization solvers: www.mcs.anl.gov/~more/dfo/shootout.html

Their results are somewhat controversial, and there could be several projects on the topic of how to make reasonable comparisons of derivative-free methods.

Tim Kelley at North Carolina State has done interesting work on what he calls “implicit filtering”, including algorithms that do not use derivatives. The website www4.ncsu.edu/~ctk/iffco.html gives a pointer to Tim’s book on the subject, and to software for noisy problems.

5. Semidefinite and convex optimization.

Semidefinite optimization (sometimes called semidefinite programming) involves optimizing a linear function in which the “variables” are symmetric positive semidefinite matrices, subject to linear constraints. (In some sense, semidefinite programming is a generalization of linear programming.) There was a flurry of activity in SDP a few years ago, but this has slowed down to some extent. A project could look at the highlights of the current state of the art in semidefinite programming. A website with many useful pointers to papers and workshops is: www-user.tu-chemnitz.de/~helmberg/semidef.html

Stephen Boyd at Stanford, one of the early leaders in semidefinite programming, has developed (with Michael Grant) CVX, a Matlab-based modeling system for convex optimization; see cvxr.com/cvx/

www.stanford.edu/~boyd/software.html

(Boyd and Grant received the Beale–Orchard-Hays Prize for Excellence in Computational Mathematical Programming at the 2012 Berlin International Symposium on Mathematical Programming.) The above links suggest numerous projects that would address a specific problem class.


The fields of optimization and machine learning are closely connected, although the exact nature of the connections is sometimes (very) obscure. This project would consider some of the optimization methods commonly used in machine learning and describe their relationship to mainstream optimization techniques, ideally suggesting ways in which there could be increased connections. Three places to start are:
  jmlr.csail.mit.edu/papers/v7/MLOPT-intro06a.html
• the papers presented at the NIPS 2008 workshop, “Optimization for machine learning”:
  opt2008.kyb.tuebingen.mpg.de/
• a special issue of the journal *Mathematical Programming* on “Optimization and Machine Learning” (volume 127, number 1, March 2011):
  www.springerlink.com/content/0025-5610/127/1/

7. Compressed sensing.
A current “hot topic” in the mathematical sciences is **compressive sensing**, which involves modern statistics as well as large-scale convex optimization. One of the leading figures in compressed sensing is Emmanuel Candès (Stanford), whose website

www-stat.stanford.edu/~candes/

provides links to his publications as well as to an interesting story from *Wired*.

A few years ago, “the Dantzig selector” (an approach obviously related to linear programming and the simplex method) was presented. The following two websites consist of a paper on this subject and a subsequent “discussion” (this is standard in the statistics literature):

www.jstor.org/stable/25464592

8. Matlab toolboxes related to optimization.
Matlab provides several optimization toolboxes, which are available when using Matlab from your Courant account (use “help” and search on “optimization”).

The Optimization Toolbox includes standard techniques for linear and quadratic programming, unconstrained optimization, etc.

The Global Optimization Toolbox includes genetic, simulated annealing, and evolutionary algorithms. These methods are intuitively appealing and very popular, but they are viewed, for the most part, as outside mainstream optimization. A possible project would be to compare (in theory, and with numerical experiments) some “standard” and “nonstandard” methods on an interesting set of test problems.

The MIT Evolutionary Design and Optimization Group has an interesting website

groups.csail.mit.edu/EVO-DesignOpt/groupWebSite/

with links to their current projects (which may suggest projects for you).

9. PDE-constrained optimization.
In many large-scale simulations, optimization takes place in a context where the problem constraints are defined by partial differential equations (PDEs). See, for example, an invited talk from the 2008 SIAM conference on optimization that provides examples of these problems:

www.siam.org/meetings/op08/Heinkenschloss.pdf

A 2011 minicourse on PDE-constrained optimization, including detailed course notes, can be found at
and suggests several possible projects.

10. **Smoothed analysis of the simplex method.**

Dan Spielman and Shang-Hua Teng made a major contribution to complexity theory in the early 2000s with their development of “smoothed analysis”, which allows one to show that, under certain conditions, the simplex method can be viewed as a polynomial-time method. The idea of smoothed analysis has since been applied with great success to other algorithms. A project on smoothed analysis might consider how numerical testing is related to theory in judging the effectiveness of the simplex method. See