

Math 693A: Advanced Numerical Analysis

Numerical Optimization

Lecture Notes #1 — Introduction

Uduak George,
(`ugeorge@sdsu.edu`)

Department of Mathematics and Statistics
Computational Sciences Research Center
San Diego State University
San Diego, CA 92182-7720

Fall 2019

Outline

- 1 The Class — Overview
 - Literature & Syllabus
 - Grading
 - Expectations and Procedures
- 2 The Class...
 - Resources
 - Formal Prerequisites
- 3 Numerical Optimization
 - The What? Why? and How?
 - Concepts & Terms
 - Mathematical Formulation

Math 693A: Literature

“Required” — (A Modern Treatment of Numerical Optimization)

Numerical Optimization, 2nd Edition, Jorge Nocedal and Stephen J. Wright, Springer Series in Operations Research, Springer Verlag, 2006. ISBN-10: 0387303030; ISBN-13: 978-0387303031

“Required” — (Supplemental)

Class notes and class web-page.

“Optional” — (A Classic in the field; Source for class projects)

Numerical Methods for Unconstrained Optimization and Nonlinear Equations, J. E. Dennis, Jr. and Robert B. Schnabel, Classics in Applied Mathematics 16, Society for Industrial and Applied Mathematics (SIAM), 1996. ISBN 0-89871-364-1.

Math 693A: Introduction — What we will cover

- Topic-2 Unconstrained Optimization
- Topic-3 Line Search Methods
- Topic-4 Trust Region Methods
- Topic-5 Conjugate Gradient Methods
- Topic-6 Quasi-Newton Methods
- Topic-7 Large-Scale Unconstrained Optimization
- Topic-8 Calculating Derivatives
- Topic-9 Least Squares Problems
- Topic-10 Nonlinear Equations

Math 693A: Introduction — Grading etc.

- 50% Homework: both theoretical, and implementation (programming) — C/C++ or Fortran are the recommended languages, but feel free to program in 6510 assembler, Java, or Matlab...
- 50% Project: Implementation of several interacting parts of an optimization package. By the end of the semester you should have a working “toolbox” of optimization algorithms which will be useful in your current and future research projects. [Complete details TBA].

Expectations and Procedures, I

- Class attendance is HIGHLY RECOMMENDED —

Expectations and Procedures, I

- Class attendance is HIGHLY RECOMMENDED —
- Class attendance is **MANDATORY** for ALL in-class presentations.

Expectations and Procedures, I

- Class attendance is **HIGHLY RECOMMENDED** —
- Class attendance is **MANDATORY** for ALL in-class presentations.
- Homework and announcements will be posted on the class web page;
If/when you attend class:

Expectations and Procedures, I

- Class attendance is **HIGHLY RECOMMENDED** —
- Class attendance is **MANDATORY** for ALL in-class presentations.
- Homework and announcements will be posted on the class web page;
If/when you attend class:
 - Please be on time.

Expectations and Procedures, I

- Class attendance is **HIGHLY RECOMMENDED** —
- Class attendance is **MANDATORY** for ALL in-class presentations.
- Homework and announcements will be posted on the class web page;
If/when you attend class:
 - Please be on time.
 - Please pay attention.

Expectations and Procedures, I

- Class attendance is **HIGHLY RECOMMENDED** —
- Class attendance is **MANDATORY** for ALL in-class presentations.
- Homework and announcements will be posted on the class web page;
If/when you attend class:
 - Please be on time.
 - Please pay attention.
 - Please turn off mobile phones.

Expectations and Procedures, I

- Class attendance is HIGHLY RECOMMENDED —
- Class attendance is **MANDATORY** for ALL in-class presentations.
- Homework and announcements will be posted on the class web page;
If/when you attend class:
 - Please be on time.
 - Please pay attention.
 - Please turn off mobile phones.
 - Please be courteous to other students and the instructor.

Expectations and Procedures, I

- Class attendance is **HIGHLY RECOMMENDED** —
- Class attendance is **MANDATORY** for ALL in-class presentations.
- Homework and announcements will be posted on the class web page;
If/when you attend class:
 - Please be on time.
 - Please pay attention.
 - Please turn off mobile phones.
 - Please be courteous to other students and the instructor.
 - Abide by university statutes, and all applicable local, state, and federal laws.

Expectations and Procedures, II

- Please, turn in assignments on time. **Assignments that are more than 1 week late will be worth 80% of the total point**(The instructor reserves the right not to accept late assignments.)

Expectations and Procedures, II

- Please, turn in assignments on time. **Assignments that are more than 1 week late will be worth 80% of the total point**(The instructor reserves the right not to accept late assignments.)
- The instructor will make special arrangements for students with documented learning disabilities and will try to make accommodations for other unforeseen circumstances, e.g. illness, personal/family crises, etc. in a way that is fair to all students enrolled in the class. **Please contact the instructor EARLY regarding special circumstances.**

Expectations and Procedures, II

- Please, turn in assignments on time. **Assignments that are more than 1 week late will be worth 80% of the total point**(The instructor reserves the right not to accept late assignments.)
- The instructor will make special arrangements for students with documented learning disabilities and will try to make accommodations for other unforeseen circumstances, e.g. illness, personal/family crises, etc. in a way that is fair to all students enrolled in the class. **Please contact the instructor EARLY regarding special circumstances.**
- Students are expected **and encouraged** to ask questions in class!

Expectations and Procedures, II

- Please, turn in assignments on time. **Assignments that are more than 1 week late will be worth 80% of the total point**(The instructor reserves the right not to accept late assignments.)
- The instructor will make special arrangements for students with documented learning disabilities and will try to make accommodations for other unforeseen circumstances, e.g. illness, personal/family crises, etc. in a way that is fair to all students enrolled in the class. **Please contact the instructor EARLY regarding special circumstances.**
- Students are expected **and encouraged** to ask questions in class!
- Students are expected **and encouraged** to to make use of office hours! If you cannot make it to the scheduled office hours: contact the instructor to schedule an appointment!

Expectations and Procedures, III

- Missed midterm exams: Don't miss exams! The instructor reserves the right to schedule make-up exams, make such exams oral presentation, and/or base the grade solely on other work (including the final exam).

Expectations and Procedures, III

- Missed midterm exams: Don't miss exams! The instructor reserves the right to schedule make-up exams, make such exams oral presentation, and/or base the grade solely on other work (including the final exam).
- Missed final exam/presentation: Don't miss the final! Contact the instructor ASAP or a grade of WU or F will be assigned.

Expectations and Procedures, III

- Missed midterm exams: Don't miss exams! The instructor reserves the right to schedule make-up exams, make such exams oral presentation, and/or base the grade solely on other work (including the final exam).
- Missed final exam/presentation: Don't miss the final! Contact the instructor ASAP or a grade of WU or F will be assigned.
- **Academic honesty**: submit your own work — but feel free to discuss homework with other students in the class! Cite any and all sources (outside of class material) you use.

Math 693A: Computer Resources

You need access to a computing environment in which to write your code; — you may want to use a combination of Matlab (for quick prototyping and short homework assignments) and C/C++ or Fortran (or something completely different).

Check out <http://julialang.org/>

Free C/C++ (gcc) and Fortran (f77, f95) compilers are available for Linux/UNIX.

You may also want to consider buying the student version of Matlab:
<http://www.mathworks.com/>

SDSU students can download a copy of matlab from
<http://edoras.sdsu.edu/~download/matlab.html>

Math 693A: Introduction — What you should know already

Math 524 and (Math 542 or Math 543)

524 \Rightarrow **Linear Algebra**

- Vector spaces, linear transformations, orthogonality, eigenvalues and eigenvectors, normal forms for complex matrices, positive definite matrices.

542 \Rightarrow **Numerical Solutions of Differential Equations**

- Initial and boundary value problems for ODEs. PDEs. Iterative methods, finite difference methods, the method of lines.

543 \Rightarrow **Numerical Matrix Analysis**

- Gaussian elimination, LU-factorizations and pivoting strategies. Direct and iterative methods for linear systems. Iterative methods for diagonalization and eigensystem computation. Tridiagonal, Hessenberg, and Householder matrices. The QR algorithm.

Math 693A: Introduction — Why???

Q: Why do we need numerical optimization methods?

A: Many problems in real life are formulated as optimization problems:

- Trajectories for airplanes, space craft, robotic motion, etc.
- Shapes for cars, airfoils, aerodynamic bicycle wheels, etc.
- Risk management — investment portfolios; insurance premiums.
- Circuit and network design.

Numerical Optimization: Concepts and Term

1 of 2

Students optimize: minimize study time T , such that GPA is acceptable. (:-)

Nature optimizes: Physical systems settle in a state of minimal energy —

- A ball rolls down to the bottom of a slope;
- DNA molecules fold to minimize some measure of energy;
- Light rays follow the path that minimizes travel time.
- Chemical reactions are energy-driven, etc, etc, etc...

In order to understand physical (economic, etc.) systems we must optimize: — first we must identify the **objective** (the measure of performance, or “energy”). The objective depends on a number of **variables** (the characteristics of the system).

Numerical Optimization: Concepts and Term

2 of 2

Our goal is to find the values of the **variables** that optimize (either minimize, or maximize) the **objective**. Often the variables are **constrained** (restricted) in some way (e.g. densities, and interest rates are non-negative).

The process of identifying the **objective**, **variables**, and **constraints** is non-trivial and will essentially be completely ignored in this class. (See **Mathematical Modeling**)

Our discussion starts after the modeling is done!

Mathematical Formulation

1 of 2

From the point of view of a mathematician, optimization is the minimization (or maximization) of a function subject to constraints on its variables.

Notation:

$\bar{\mathbf{x}}$ *the vector of variables (a.k.a. unknowns, or parameters)*
 $f(\bar{\mathbf{x}})$ *the objective function*
 $\bar{\mathbf{c}}$ *the vector of constraints that the unknowns must satisfy*

The **Optimization Problem** can be written

$$\min_{\bar{\mathbf{x}} \in \mathbb{R}^n} f(\bar{\mathbf{x}}) \quad \text{subject to} \quad \begin{cases} c_i(\bar{\mathbf{x}}) = 0, & i \in E \\ c_i(\bar{\mathbf{x}}) \geq 0, & i \in I \end{cases}$$

Mathematical Formulation

2 of 2

The **Optimization Problem**

$$\min_{\bar{\mathbf{x}} \in \mathbb{R}^n} f(\bar{\mathbf{x}}) \quad \text{subject to} \quad \begin{cases} c_i(\bar{\mathbf{x}}) = 0, & i \in E \\ c_i(\bar{\mathbf{x}}) \geq 0, & i \in I \end{cases}$$

Here E is the set of **equality constraints**, and I the set of **inequality constraints**.

Note that a maximization problem can be converted into a minimization problem:

$$\max_{\bar{\mathbf{x}} \in \mathbb{R}^n} f(\bar{\mathbf{x}}) \quad \Leftrightarrow \quad \min_{\bar{\mathbf{x}} \in \mathbb{R}^n} [-f(\bar{\mathbf{x}})]$$

and a less-than-or-equal-to constraint can similarly be converted into a greater-than-or-equal-to constraint.

Feasible Region

The set of all $\bar{\mathbf{x}} \in \mathbb{R}^n$ which satisfy the constraints $\bar{\mathbf{c}}$ is called the **feasible region**, e.g. if

$$\begin{aligned}c_1(x_1, x_2) &= x_1^2 + x_2^2 \geq 1 \\c_2(x_1, x_2) &= -(x_1^2 + x_2^2) \geq -4\end{aligned}$$

then the feasible region is the annulus:

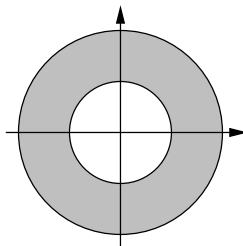


Figure: The annulus $1 \leq r \leq 2$.

Constrained vs. Unconstrained Optimization

1 of 2

Optimization problems of the form

$$\min_{\bar{\mathbf{x}} \in \mathbb{R}^n} f(\bar{\mathbf{x}}) \quad \text{subject to} \quad \begin{cases} c_i(\bar{\mathbf{x}}) = 0, & i \in E \\ c_i(\bar{\mathbf{x}}) \geq 0, & i \in I \end{cases}$$

can be classified according to the nature of the function and constraints (linear, non-linear, convex, etc.) — the key distinction is between problems that have constraints, and problems that do not:

Constrained Optimization Problems: arise from models that include explicit constraints on the variables. They can be relatively simple, or nasty non-linear inequalities expressing complex relationships between the variables.

Constrained vs. Unconstrained Optimization

2 of 2

Unconstrained Optimization Problems arise directly in some applications; if the constraints are “natural” it may be safe to disregard them during the solution process and verify that they are satisfied in the solution.

Further, constrained problems can be restated as unconstrained problems — the constraints are replaced by penalizing terms in the objective which “discourage” violation of the constraints.

The more complicated the constraints, the more difficult it is to find the optimal solution. The absence of constraints is the easiest case.

Continuous vs. Discrete Optimization

In many applications, the variables (\bar{x}) can only take integer values — very few people would be interested in buying $3/4$ of a TV set, or receive $1/3$ of a package; the electrons in an atom can only exist in certain quantum states, etc, etc.

The Discrete Optimization Problem is harder than the Continuous Optimization Problem. — One way to get “close” to solving the discrete problem is to solve the problem as if it is continuous and then round or truncate the solution to integer values. This will often give a sub-optimal integer solution and/or a solution that is **infeasible**.

Here we will only consider the “easier” Continuous Optimization Problem.

Global and Local Optimization

The fastest optimization algorithms seek a **local solution** — a point where the objective is smaller than all other feasible points in its vicinity.

The best solution — the **global minimum** is usually hard to find, but is often desirable.

Under certain circumstances (e.g. see convexity) there is only one minimum.

We will focus on local optimization algorithms, but note that (most) global algorithms will solve a sequence of local optimization problems.

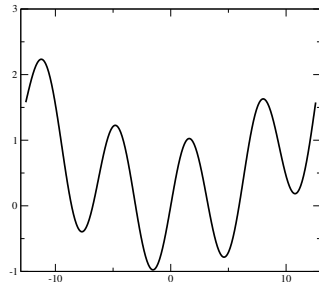
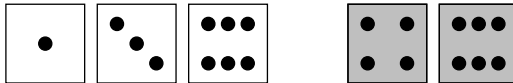


Figure: A function with multiple local minima, and one global minimum.

Stochastic and Deterministic Optimization



In many applications it is impossible to fully specify all parameters at the time of formulation; in quantum physics, the stock market, or the game of “risk” some quantities are “random” and are best modeled using some probability model.

We will focus on **deterministic optimization** problems, where the model can be fully specified when we formulate the problem.

However, in many cases the solutions to **stochastic optimization** problems are formulated as sequences or collections of deterministic problems.

Convexity: Definitions

There are two “types” of convexity which impact optimization problems:

- $S \subseteq \mathbb{R}^n$ is a **convex set** if the straight line segment connecting any two points in S lies entirely inside S . Formally, for any two points $\bar{\mathbf{x}} \in S$ and $\bar{\mathbf{y}} \in S$, we have $(\alpha\bar{\mathbf{x}} + (1 - \alpha)\bar{\mathbf{y}}) \in S$ for all $\alpha \in [0, 1]$.
- f is a **convex function** if its domain is a convex set and if for any two points $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ in this domain, the graph of f lies below the straight line connecting $(\bar{\mathbf{x}}, f(\bar{\mathbf{x}}))$ to $(\bar{\mathbf{y}}, f(\bar{\mathbf{y}}))$ in \mathbb{R}^{n+1} . That is, we have

$$f(\alpha\bar{\mathbf{x}} + (1 - \alpha)\bar{\mathbf{y}}) \leq \alpha f(\bar{\mathbf{x}}) + (1 - \alpha)f(\bar{\mathbf{y}}), \quad \forall \alpha \in [0, 1]$$

Convexity: Illustrations

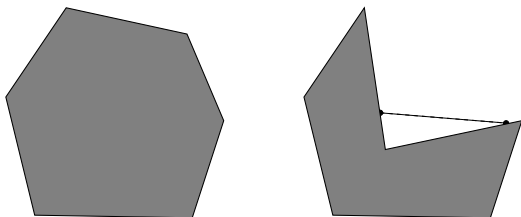


Figure: A convex (left) and a non-convex set (right) in \mathbb{R}^2 .

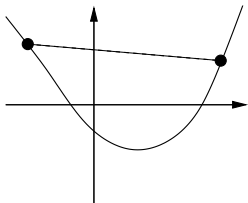


Figure: A convex function.

Convexity: Notes

A function f is said to be **concave** if $-f$ is convex.

Optimization algorithms for unconstrained problems are usually guaranteed to converge to a stationary point (maximum, minimum, or inflection point) of the objective f .

If f is convex, **then** the algorithm has converged to a global optimum.

Bottom line: Convexity simplifies the problem.

Summary

Easier	Harder
<i>Unconstrained</i>	<i>Constrained</i>
<i>Continuous</i>	<i>Discrete</i>
<i>Local Optimization</i>	<i>Global Optimization</i>
<i>Deterministic</i>	<i>Stochastic</i>
<i>Convex</i>	<i>Non-Convex</i>

Table: Summary of some factors impacting the difficulty of the optimization problem.

In this class we will mainly look at Local Optimization methods for Deterministic, Unconstrained, Continuous, Convex functions over Convex sets. Still, it will be a challenging semester!

Algorithms

Optimization algorithms are **iterative** and generate a sequence of successively better estimates of the solution.

Three key attributes characterize each (good) algorithm:

Robustness: Algorithm performance on a wide variety of problems (of the same type), for a range of reasonable choices of initial values.

Efficiency: We prefer fast algorithms that do not require excessive amounts of storage.

Accuracy: The algorithm should find the solution without being overly sensitive to errors in the data or roundoff errors in the computations.

These goals are often **conflicting** — hence careful consideration of trade-off between the goals is a key part of this course.

Index

Course Literature, 10

Grading

Homework, 12

Late policy, 23

Project, 12

Matlab

Free download, 33

Optimization

Mathematical formulation, 38

Prerequisites, 34