Topics in Numerical
and Computational Mathematics

Computational Optimization:

Success in Practice

Chapter 1: Introduction to Optimization

VladislavBukshtynov@yahoo.com

CRC Press
https://www.crcpress.com/

Optimization Models

Main goal:

1. formulate a model in terms of mathematical notations, and
2. find the set of parameters for this model attempting to find the best possible solution for the problem the model is created for

Model complexity:

- model size
- simplicity to find accurate mathematical description

Examples:

- Models to describe medical, biological, chemical processes.
- Better knowledge of natural objects to minimize loses from natural disasters & maximize profit from its power.
- Design of the business structures to minimize loses and maximize profit.
- Geometry of airfoil/rocket/submarine to minimize weight/drag and to maximize lift.
- Other examples from your own research and expertise area(s).

General Notations for Optimization Problem

$$\begin{aligned} \min_{\mathbf{u}\in\mathbb{R}^n} / \max \quad & f(\mathbf{u}) \\ \text{subject to} \quad & h_i(\mathbf{x};\mathbf{u}) = 0, \qquad i = 1, \ldots, p \\ & g_j(\mathbf{x};\mathbf{u}) \leq 0, \qquad j = 1, \ldots, m \end{aligned}$$

- min/max problem
- optimization (control, decision, design) variable $\mathbf{u} \in \mathbb{R}^n$
  - state $\mathbf{x}$ vs. control $\mathbf{u}$ variables
- objective (cost) function(al) $f(\mathbf{u}) : \mathbb{R}^n \to \mathbb{R}$ (scalar)
  - in general we consider $f(\mathbf{x};\mathbf{u})$
- constraints
  - equality/inequality
  - ODE/PDE systems
  - other requirements, e.g. functional spaces for $\mathbf{x}$ and $\mathbf{u}$
- feasible region (feasible set, control/solution space) $\mathbb{S}$, defined by constraints
  - feasible $\mathbf{u} \in \mathbb{S}$ vs. infeasible $\mathbf{u} \notin \mathbb{S}$ solutions
- optimal solution $\mathbf{u}^* = \underset{\mathbf{u}\in\mathbb{S}}{\operatorname{argmin}} f(\mathbf{u})$

Classification of Optimization Problems

- type of constraints: unconstrained/constrained (ODE/PDE-based optimization)
- nature of equations involved: linear programming (LP) & nonlinear programming (NLP) problems, quadratic (QP), etc.
- permissible values of controls: continuous, discrete (e.g. integer programming)
- deterministic nature of variables: deterministic, stochastic (or probabilistic)
- separability of functions: separable, non-separable
- number of objectives: single-objective, multi-objective
- other types: optimal control, non-optimal control, etc.

Example 1.1: Constrained Optimization with Nonlinear Objective

Find the point $\mathbf{x} = [x_1 \ x_2]^T \in \mathbb{R}^2$ on the line $x_1 + x_2 = 10$ closest to the point $(10, 10)$

$$d = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} \rightarrow \text{minimize}$$

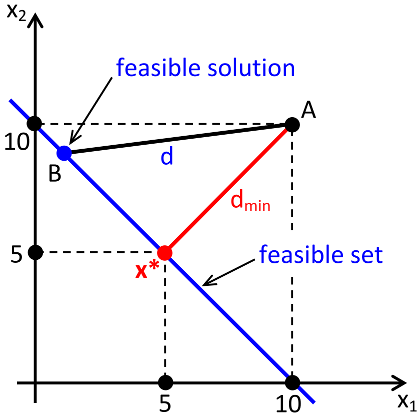
Problem: constrained 2D QP optimization

$$\min_{\mathbf{x} \in \mathbb{R}^2} \quad f(\mathbf{x}) = (x_1 - 10)^2 + (x_2 - 10)^2$$

$$\text{s.t.} \quad x_1 + x_2 = 10$$

Solution: optimal $\mathbf{x}^* = [5 \ 5]^T$ found

- geometrically (in figure)
- analytically (goes to homework)

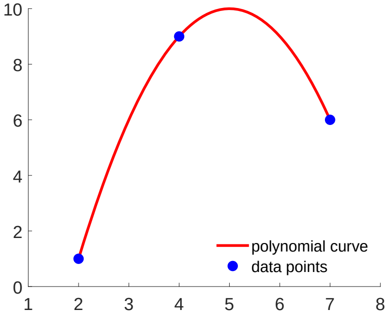


Example 1.2: Data Fitting

Data: 3 points $(2,1)$, $(4,9)$, $(7,6)$
Find coefficients a_1, a_2, a_3 assuming quadratic fit $y = a_1 + a_2 x + a_3 x^2$

Solution (exact): $\mathbf{a} = [-15\ 10\ \ -1]^T$ or $y(x) = -15 + 10x - x^2$ found from the system of linear equations

$$\begin{cases} a_1 + 2a_2 + 4a_3 & = 1, \\ a_1 + 4a_2 + 16a_3 & = 9, \\ a_1 + 7a_2 + 49a_3 & = 6. \end{cases}$$



Using matrix notation:

$$A\,\mathbf{a} = \mathbf{y} \quad \Rightarrow \quad \mathbf{a} = A^{-1}\mathbf{y},$$

where

$$A = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 4 \\ 1 & 4 & 16 \\ 1 & 7 & 49 \end{bmatrix}, \qquad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}, \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 9 \\ 6 \end{bmatrix}$$

solvable (conditionally): # of data $=$ # of unknowns; what are the conditions?

Q: what if the problem is overdetermined ($>$) / underdetermined ($<$)?

Example 1.2: Data Fitting (cont'd)

MATLAB code: `Chapter_1_data_fit.m`

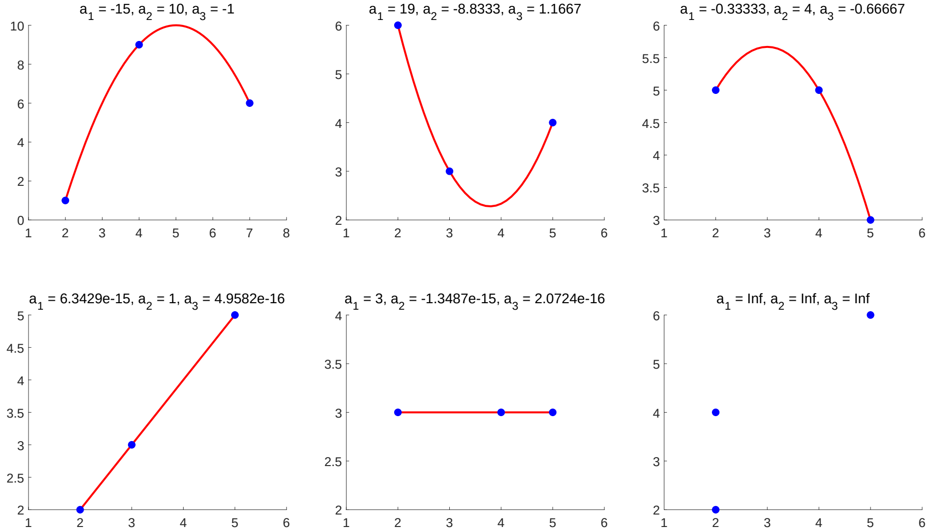
- input data: `data = [2 1; 4 9; 7 6];`
- solution #1: by separate m-code `DataFitM.m`
- solution #2: by user-defined m-function `DataFitFn`
- solution #3: by Matlab's built-in function `POLYFIT(X,Y,N)`
- solution conversion: polynomial preparation
- visualization

Aspects to consider:

- implementation: m-code vs. user-defined function vs. built-in function
- code readability: block-wise structures, comments, self-explanatory naming, etc.
- user-defined interface to prevent code breaking
- proper debugging with multiple tests & special cases (next slide)
- visualization to comfort analysis of the results

Example 1.2: Data Fitting (cont'd)

Test cases:



Example 1.3: Least-Squares Data Fitting

Consider previous example: new (4th) data point $(3, 6)$ fits the model

If $y_4 \neq 6$ or other data points (measurements) suffer from errors $\rightarrow$ model equation $y = a_1 + a_2 x + a_3 x^2$ cannot be solved exactly!

Residual vector for m "pieces" of data

$$\mathbf{r} = \mathbf{y} - A\,\mathbf{a} = \begin{bmatrix} y_1 - (a_1 + a_2 x_1 + a_3 x_1^2) \\ y_2 - (a_1 + a_2 x_2 + a_3 x_2^2) \\ \cdots \\ y_m - (a_1 + a_2 x_m + a_3 x_m^2) \end{bmatrix}$$

Least-squares data fitting: (common approach)

$$\min_{\mathbf{a} \in \mathbb{R}^3} f(\mathbf{a}) = r_1^2 + r_2^2 + \cdots + r_m^2 = \sum_{i=1}^{m} \Big(y_i - (a_1 + a_2 x_i + a_3 x_i^2) \Big)^2,$$

where $\mathbf{a} = [a_1\ a_2\ a_3]^T$.

- $(2, 1), (4, 9), (7, 6), (3, 6)$ - "perfect match" $\rightarrow \mathbf{r} = \mathbf{0}$
- otherwise $\rightarrow \mathbf{r} \neq \mathbf{0}$, where r_i describes data "mismatch", $i = 1, \ldots, m$

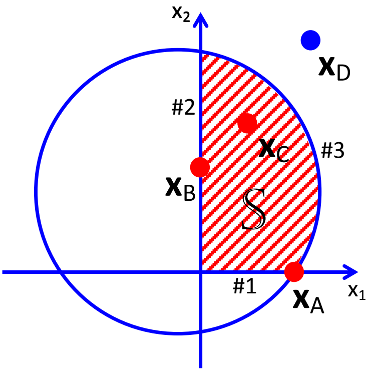
Feasibility

Constrained optimization problem (simplified, state $\mathbf{x}$ is also control):

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x})$$

$$\boxed{\begin{aligned} \text{s.t.} \quad & h_i(\mathbf{x}) = 0, \qquad i = 1, \dots, p \\ & g_j(\mathbf{x}) \le 0, \qquad j = 1, \dots, m \end{aligned}} \qquad (*)$$

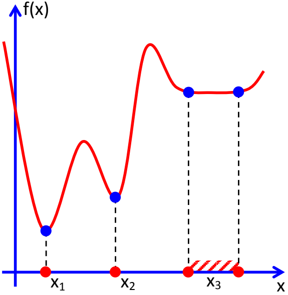
- feasible region (set) $\mathbb{S}$: a set of all solutions satisfying $(*)$
- feasible solution: $\mathbf{x} \in \mathbb{S}$
- boundary of feasible region and interior points
- active constraint $g_i(\mathbf{x}) \le 0$ at $\mathbf{x}_0$: if $g_i(\mathbf{x}_0) = 0$
- active set of constraints at $\mathbf{x}_0$: all active constraints
- eliminating constrains by adding them to objective (substitution, Lagrange multipliers)

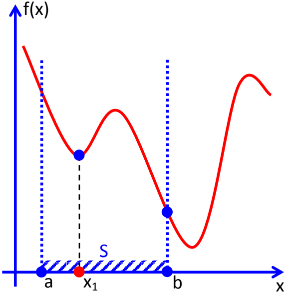


Optimality

Consider general optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x})$$
$$\text{s.t.} \quad \mathbf{x}\in\mathbb{S}$$

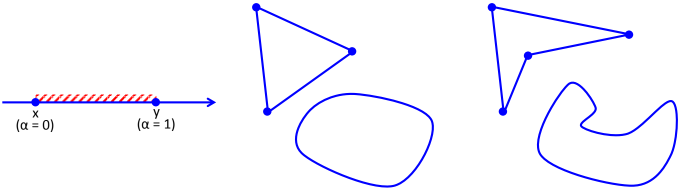




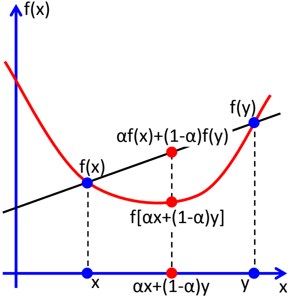
- $\mathbf{x}^*$ is a global minimizer of f in $\mathbb{S}$: if $\forall \mathbf{x}\in\mathbb{S} \quad f(\mathbf{x}^*) \le f(\mathbf{x})$
- $\mathbf{x}^*$ is a local minimizer of f in $\mathbb{S}$: if $\forall \mathbf{x}\in\mathbb{S} \quad \text{s.t. } \|\mathbf{x}-\mathbf{x}^*\| < \epsilon \quad f(\mathbf{x}^*) \le f(\mathbf{x})$
- in case $f(\mathbf{x}^*) < f(\mathbf{x})$, but $\mathbf{x}\neq\mathbf{x}^*$, $\mathbf{x}^*$ is a strict local (global) minimizer

Convexity (convex set vs. convex function)

A **set** $\mathbb{S}$ is convex if $\quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{S}: \quad \alpha\mathbf{x} + (1-\alpha)\mathbf{y} \in \mathbb{S}, \quad \forall\ 0 \leq \alpha \leq 1$



(a) 1D: convex (b) 2D: convex (c) 2D: non-convex



A **function** f is convex on a convex set $\mathbb{S}$ if $\quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{S}:$

$$f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha)f(\mathbf{y})$$

$$\forall\ 0 \leq \alpha \leq 1$$

- in case of $\geq$ function f is concave
- strictly convex/concave for $<$ or $>$

General Optimization Algorithm (iterative)

To solve the problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x})$$
$$\text{s.t.} \quad \mathbf{x} \in \mathbb{S}$$

1. Choose initial guess $\mathbf{x}^0$ (and algorithm settings)
2. For $k = 1, 2, \ldots$ check (computational) optimality of $\mathbf{x}^k$, e.g.
   - $\mathbf{x}^k$ reduces $f(\mathbf{x})$ up to a necessary level
   - $\boldsymbol{\nabla} f(\mathbf{x}^k) \cong \mathbf{0}$ (local optimum condition)
   - other termination conditions (next slide)
3. If OK (optimal) $\to$ **STOP**
4. Find solution update (change, perturbation) $\boldsymbol{\delta}^k$,
   e.g. by means of search direction $\mathbf{d}^k$
5. Update the solution
$$\mathbf{x}^{k+1} = \mathbf{x}^k + \boldsymbol{\delta}^k = \mathbf{x}^k + \alpha^k \mathbf{d}^k$$
6. Go to 2

- search direction $\mathbf{d}^k$ improves the solution in some sense, e.g. $\mathbf{d}^k = -\boldsymbol{\nabla} f(\mathbf{x}^k)$
- step size α^k is determined in assumption $f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$

General Optimization Algorithm (cont'd): Termination Conditions

Based on

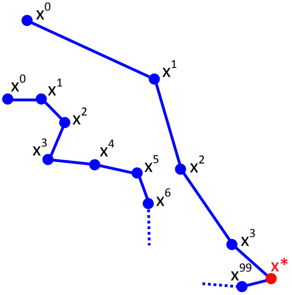
1. sufficient changes in the solution: (in some norm N)
   - absolute decrease $\left\|\mathbf{x}^k - \mathbf{x}^{k-1}\right\|_N < \epsilon$
   - relative decrease $\dfrac{\left\|\mathbf{x}^k - \mathbf{x}^{k-1}\right\|_N}{\left\|\mathbf{x}^{k-1}\right\|_N} < \epsilon$

2. sufficient changes in the objective:
   - absolute decrease $\left|f(\mathbf{x}^k) - f(\mathbf{x}^{k-1})\right| < \epsilon$
   - relative decrease $\left|\dfrac{f(\mathbf{x}^k) - f(\mathbf{x}^{k-1})}{f(\mathbf{x}^{k-1})}\right| < \epsilon$ (★)

3. computational efforts:
   - max number of optimization iterations k_{max}: $k > k_{\text{max}}$ (★)
   - max number of objective evaluations
   - limit on elapsed computational time T: $t > T$

Q: (★) are recommended options (problem dependent). Why?

Convergence



Computational complexity: # of arithmetic operations required to find a solution

Solving a problem iteratively: does it converge? if yes, how fast?

Sequence of errors $\mathbf{e}^k = \mathbf{x}^k - \mathbf{x}^*$ assuming $\lim_{k\to\infty} \mathbf{e}^k = \mathbf{0}$

Sequence $\{\mathbf{x}^k\}$ **converges** to $\mathbf{x}^*$ $\left(\{\mathbf{x}^k\} \to \mathbf{x}^*\right)$ with rate r and constant C if

$$\lim_{k\to\infty} \frac{\|\mathbf{e}^{k+1}\|}{\|\mathbf{e}^k\|^r} = \lim_{k\to\infty} \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|^r} = C, \quad C < \infty$$

- linear convergence: $\|\mathbf{e}^{k+1}\| = C\|\mathbf{e}^k\|,\ r = 1$
  - $0 < C < 1$ – converges ($C \to 0$ faster, $C \to 1$ slower)
  - $C > 1$ – diverges
- superlinear: $C = 0$, also $r > 1$
- sublinear: $C = 1$
- quadratic $r = 2$; cubic $r = 3$; ...
- hard to analyze in real computations as $\mathbf{x}^*$ is not always available

Homework for Chapter 1

- Solve problem in Example 1.1 analytically.
- Modify MATLAB code `Chapter_1_data_fit.m` for Example 1.2:
  - to work with new data (4 or 5 points) by using all 3 solution approaches,
  - repeat for data with m ($m > 5$) points,
  - implement check-up to prevent code breaking in case the problem is under- or overdetermined,
  - implement check-up to prevent inf/nan problem in case the data is "defective".
- Show that a set is convex if and only if its intersection with any line is convex.
- In general the product or ratio of two convex functions is not convex. However, there are some results that apply to functions on $\mathbb{R}$. Prove the following.
  - If f and g are convex, both nondecreasing (or nonincreasing), and positive functions on an interval, then fg is convex.
  - If f is convex, nondecreasing, and positive, and g is concave, nonincreasing, and positive, then f/g is convex.

Homework for Chapter 1 (cont'd)

- Using second-order condition for convexity for each of the following functions determine whether it is convex or concave:
  (a) $f(x) = e^x - 1$ on $\mathbb{R}$,
  (b) $f(x_1, x_2) = x_1 x_2$ on $\mathbb{R}_+ \times \mathbb{R}_+$,
  (c) $f(x_1, x_2) = \dfrac{1}{x_1 x_2}$ on $\mathbb{R}_+ \times \mathbb{R}_+$,
  (d) $f(x_1, x_2) = \dfrac{x_1^2}{x_2}$ on $\mathbb{R} \times \mathbb{R}_+$,
  (e) $f(x_1, x_2) = x_1^{\alpha} x_2^{1-\alpha}$, where $0 \le \alpha \le 1$, on $\mathbb{R}_+ \times \mathbb{R}_+$.
- For each of the following sequences with given general term x^k, prove that the sequence converges, find its limit, and determine convergence parameters r and C:
  (a) $x^k = 2^{-k}$,
  (b) $x^k = 1 + 5 \cdot 10^{-2k}$,
  (c) $x^k = 3^{-k^2}$.

Where to Read More for Chapter 1

- **Bukshtynov (2023):** Chapter 1
- **Bertsimas (1997):** Chapter 1 (Introduction), Chapter 2 (The Geometry of Linear Programming)
- **Boyd (2004):** Chapter 1 (Introduction), Chapter 2 (Convex Sets), Chapter 3 (Convex Functions)
- **Griva (2009):** Chapter 1 (Optimization Models), Chapter 2 (Fundamentals of Optimization)
- **Nocedal (2006):** Chapter 1 (Introduction), Chapter 2 (Fundamentals of Unconstrained Optimization)

MATLAB codes for Chapter 1

- `Chapter_1_data_fit.m`
- `DataFitM.m`