Linearly Constrained Optimization

January 1, 2023

Linearly Constrained Optimization

General Formulation

$$\min_{x\in\mathbb{R}^n} \psi(\mathbf{x}), \;\text{s.t.}\; \mathbf{b}_l \le \mathbf{A}\mathbf{x} \le \mathbf{b}_u \text{ and } \mathbf{l} \le \mathbf{x} \le \mathbf{u} \quad (1)$$

- $\psi(\mathbf{x})$ is a general smooth (twice continuously differentiable) function with gradient $\mathbf{g}(\mathbf{x})$ and Hessian $\mathbf{H}(\mathbf{x})$.
- $\mathbf{b}_l \le \mathbf{A}\mathbf{x} \le \mathbf{b}_u$ are general linear constraints.
- $\mathbf{l} \le \mathbf{x} \le \mathbf{u}$ are simple bound constraints.
- Mathwrist takes the general form (1) and solves it using active set method.
- without loss of generality, our discussion assumes a convenient form

$$\min_{x\in\mathbb{R}^n} \psi(\mathbf{x}), \;\text{s.t.}\; \mathbf{A}\mathbf{x} \ge \mathbf{b} \quad (2)$$

- a special case of (1) is box-constrained optimization. Mathwrist has a dedicated solver for it.

Linearly Constrained Optimization

Active Set Method

- same framework discussed in our linear programming (LP) and quadratic programming (QP) documentation.
- retain and apply economy update of **QR** factoring,

$$\mathbf{A}_{\mathcal{W}}^T = \underbrace{\left(\mathbf{Y} \mid \mathbf{Z} \right)}_{\mathbf{Q}} \begin{pmatrix} \mathbf{R} \\ 0 \end{pmatrix}$$

- iteratively make step moves along a descent null space direction $\mathbf{p} = \mathbf{Z}\mathbf{p}_z$ wrt the current active set $\mathcal{W}$.
- same techniques as unconstrained optimization to compute $\mathbf{p}_z$.

Linearly Constrained Optimization

Taylor expansion

For any search direction $\mathbf{p}$ and step length $\hat{\alpha}$,

$$\psi(\mathbf{x}_k + \hat{\alpha}\mathbf{p}) = \psi(\mathbf{x}_k) + \hat{\alpha}\mathbf{p}^T\mathbf{g}(\mathbf{x}_k) + \frac{1}{2}\hat{\alpha}^2\mathbf{p}^T\mathbf{H}(\hat{\mathbf{x}})\mathbf{p}$$

, where

$$\hat{\mathbf{x}} = \mathbf{x}_k + \hat{\alpha}\theta\mathbf{p}, 0 \leq \theta \leq 1$$

In a null space direction, $\mathbf{p} = \mathbf{Z}\mathbf{p}_z$, the step reduction

$$\Delta\psi_k = \psi(\mathbf{x}_k + \hat{\alpha}\mathbf{p}) - \psi(\mathbf{x}_k) = \hat{\alpha}\mathbf{p}_z^T\tilde{\mathbf{g}}(\mathbf{x}_k) + \frac{1}{2}\hat{\alpha}^2\mathbf{p}_z^T\tilde{\mathbf{H}}(\hat{\mathbf{x}})\mathbf{p}_z \quad (3)$$

$$\tilde{\mathbf{g}}(\mathbf{x}_k) = \mathbf{Z}^T\mathbf{g}(\mathbf{x}_k) \text{ is the reduced gradient} \quad (4)$$

$$\tilde{\mathbf{H}}(\hat{\mathbf{x}}) = \mathbf{Z}^T\mathbf{H}(\hat{\mathbf{x}})\mathbf{Z} \text{ is the reduced Hessian} \quad (5)$$

Linearly Constrained Optimization

Line Search

- for a descent direction, we need the first order term $\mathbf{p}_z^T \tilde{\mathbf{g}}(\mathbf{x}_k) < 0$.
- the second order term in (3) will dominate $\Delta\psi_k$ for large $\hat{\alpha}$.
- if the reduced Hessian in (5) has positive curvature, $\Delta\psi_k$ overshooting for large $\hat{\alpha}$.
- for convex QP problems, unit step $\hat{\alpha} = 1$ reaches the local optimal.
- for general $\psi(\mathbf{x})$, we need a line search algorithm to determine $\hat{\alpha}$.
- actual step length $\alpha \in [0, \hat{\alpha}]$ determined by active set, i.e. by a blocking constraint.

Linearly Constrained Optimization

Search Directions

- Modified Newton method: solve $\mathbf{p}_z\tilde{\mathbf{H}}(\mathbf{x}_k) = -\tilde{\mathbf{g}}(\mathbf{x}_k)$
  - modified Cholesky on reduced Hessian $\tilde{\mathbf{H}}(\mathbf{x}_k)$.
  - apply low rank update on the factorization whenever possible.
  - when $\tilde{\mathbf{H}}(\mathbf{x}_k)$ is indefinite, compute a direction of negative curvature.
- Quasi-Newton: solve $\mathbf{p}_z\mathbf{Z}^T\mathbf{B}_k\mathbf{Z} = -\tilde{\mathbf{g}}(\mathbf{x}_k)$, where $\mathbf{B}_k$ is the approximation of Hessian $\mathbf{H}(\mathbf{x}_k)$.
  - $\mathbf{B}_k$ is obtained by BFGS update from $\mathbf{B}_{k-1}$.
  - compute reduced Cholesky $\mathbf{Z}^T\mathbf{B}_k\mathbf{Z} = \mathbf{L}\mathbf{L}^T$ to solve $\mathbf{p}_z$.

Linearly Constrained Optimization

Optimality Conditions

- $\mathbf{x}^*$ is feasible, $\mathbf{A}_\omega \mathbf{x}^* = \mathbf{b}_\omega$.
- $\mathbf{g}(\mathbf{x}^*) = \mathbf{A}_\omega^T \lambda$ or equivalently $\mathbf{Z}^T \mathbf{g}(\mathbf{x}^*) = 0$.
- $\mathbf{Z}^T \mathbf{H}(\mathbf{x}^*) \mathbf{Z}$ is positive semi-definite (necessary) or positive definite (sufficient).
- Lagrange multiplier $\lambda_i \geq 0$ (necessary) or $\lambda_i > 0$ (sufficient), for all lower bounded constraints $i \in \omega$.

References I

[1] Jorge Nocedal and Stephen J. Wright: Numerical Optimization, Springer, 1999

[2] Philip E. Gill, Walter Murray and Margaret H. Wright: Practical Optimization, Academic Press, 1981

[3] Philip E. Gill and Walter Murray: Newton-Type Methods for Unconstrained and Linearly Constrained Optimization. Mathematical Programming 7 (1974), pp. 311-350

[4] Philip E. Gill, G. H. Golub, Walter Murray and Michael. A. Saunders: Methods for Modifying Matrix Factorizations, Mathematics of Computation, Volumn 28, Number 126, April 1974, pages 505-535

[5] Philip E. Gill, Walter Murray and Michael A. Saunders: Methods for computing and modifying the LDV factors of a matrix, Mathematics of Computation, Volumn 29, Number 132, October 1975, pages 1051-1077

References II

[6] Anders. Forsgren, Philip E. Gill and Walter Murray: Computing Modified Newton Directions Using a Partial Cholesky Factorization, SIAM J. SCI. COMPUT. Vol. 16, No. 1, pp. 139-150