CS 450: Numerical Anlaysis

Lecture 18
Chapter 6 Numerical Optimization
Conjugate Gradient and Constrained Optimization

Edgar Solomonik

Department of Computer Science
University of Illinois at Urbana-Champaign

March 28, 2018

Sequential Quadratic Programming → iterative sol'ns x_k

- *Sequential quadratic programming (SQP)* solves a constrained quadratic program at the kth step:

Newton's method for the equality - constrained nonlinear optimization problem

reduce to constrained quadratic

1st: transform constrained problem to Lagrangian

2nd: approximate by quadratic → 3rd step: transform constrained quadratic

- We can reduce a constrained quadratic program to an unconstrained one, using the Lagrangian function of q:

into unconstrained QP

Solving Quadratic Programs

- Newton's method for optimization can solve the quadratic program with a single step:

after reduction, obtain unconstrained QP

Newton's method for opt., solves QP exactly

$O(n^3)$ work / SQP iteration minima of quadratic is true minima

- The *conjugate gradient method* provides an effective way of solving QPs iteratively:

iterative method, instance of Krylov subspace method

minimize $\frac{1}{2}x^T A x + c^T x$

$y = Ax \Rightarrow Af(y)$

dominant step

Parallel Tangents CG

$\hat{x}_{k+1}$ from gradient descent

line search along x_{k-1} x_k x_{k+1}

Conjugate Gradient as a Krylov Subspace Method

- Generally, Krylov subspaces describe the information available from k matrix-vector products, and can be used to find an approximation x_k to the minima of $c^T x - x^T A x$:

$\mathcal{K}(A, x_0) = \text{span}(x_0, Ax_0, \dots, A^{k-1}x_0)$

$K_k = [x_0 \ \ Ax_0 \ \dots \ A^{k-1}x_0] = Q_k R_k$

$T = Q_k^T A Q_k$ T is upper-Hessenberg

obtain x_{k+1} from x_k and x_{k-1}

- Conjugate gradient can be derived from vectors generated by the Lanczos algorithm for symmetric (positive-definite) A, yielding

$$\boxed{x_k = QT^{-1}e_1\|c\|_2} \quad \text{if} \quad x_0 = c$$

T is tridiagonal!

Lanczos:

$y_k = Aq_k = \alpha q_{k+1} + \beta q_k + \gamma q_{k-1} \Rightarrow \langle q_{k+1}, Aq_{k-1}\rangle = \alpha\langle q_{k+1}, q_{k-1}\rangle$

orthogonalize y_k w.r.t. q_k and $q_{k-1} \Rightarrow q_{k+1} + \beta \dots + \gamma \dots = 0$

q_{k+1} is A-orthogonal to q_{k-1} | $\langle q_{k+1}, Aq_{k-1}\rangle = 0$

Seek $x_k \in K_k(c) = \text{span}(c, Ac, \ldots, A^{k-1}c)$

$\min \frac{1}{2} x^T A x + c^T x = \min f(x)$

1. want to minimize $\|x^* - x_k\|_2 = \|e_k\|_2$

   - not enough information

2. instead can minimize $\left\| \overbrace{\frac{1}{2} x_k^T A x_k + c^T x_k - f(x^*)}^{r(x_k)} \right\|_2$

   MINRES GMRES

3. CG minimize $\|r(x_k)\|^2_{A^{-1}} = r(x) A^{-1} r(x)$

$$x_k = Q\,T^{-1} e_1 \|c\|_2 \qquad Q = \begin{bmatrix} \frac{c}{\|c\|_2} & \vdots & \vdots & \cdots \end{bmatrix}$$

$$\square \; \backslash\backslash\backslash^{-1} \begin{bmatrix} \|c\|_2 \\ 0 \\ \vdots \end{bmatrix}$$

$$Q^T(c - A x_k) = Q^T c - Q^T A x_k$$

$$= e_1 \|c\|_2 - \underbrace{Q^T A Q}_{T} T^{-1} e_1 \|c\|$$

$$= 0$$

Conjugate Gradient Properties

- Each iteration of conjugate gradient has cost proportional to a matrix-vector product: + inner products

- Conjugate gradient is especially efficient when the matrix has a sparse or implicit representation:

matrix A is often related to Hessian of f

H can be approximated based on values of f, suffice to scan $O(n)$ values to compute implicit Ax

Active Set Methods

- To use SQP for an inequality constrained optimization problem, consider at each iteration an *active set* of constraints:

SQP solves equality-constrained NP

pick set of active constraints,

treat them as equality constraints, and

solve SQP ignoring other constraints

problem: approximation outside feasible region

- The Karush-Kuhn-Tucker (KKT) optimality conditions in this case are

$$\nabla_{x}\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda}) = \mathbf{0}$$

$$\boldsymbol{g}(\boldsymbol{x}) = \mathbf{0}$$ ← equality cstrd.

$$\boldsymbol{h}(\boldsymbol{x}) \leq \mathbf{0}$$ ← inequality — active

$$\boldsymbol{\lambda}_2 \geq \mathbf{0}$$ ← Lagrange multipliers for $h(x)$

$$\boldsymbol{\lambda}_2^T\boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}$$

either $h_i(x^*) = 0$ or $\lambda_i = 0$

Penalty Functions

- We can reduce constrained optimization problems to unconstrained ones by modifying the objective function. *Penalty* functions are effective for equality constraints $g(x) = 0$:

$$\lim_{\rho \to \infty} x^*_\rho = x^*$$

$$\varphi_\rho(x) = f(x) + \rho \underbrace{g(x)^T g(x)}_{\text{inner product}}$$

$\rho > 1$

x^*_ρ as minima of $\varphi_\rho(x)$ and take $\rho \to \infty$

- The augmented Lagrangian function provides a more numerically robust approach:

unstable

$$\mathcal{L}_\rho(x, \lambda) = f(x) + \lambda^T g(x) + \rho\, g(x)^T g(x)$$

Barrier Functions

- A drawback of penalty function methods is that they can produce infeasible approximate solutions, which is problematic if the objective function is only defined in the feasible region:

stay within feasible region

$$\varphi_\mu(x) = f(x) + \mu \sum_{i=1}^{m} \frac{1}{h_i(x)}$$

never pass barrier → stay feasible, as $\mu \to 0$

$x_k^* \to x^*$

- Barrier functions provide an effective way (*interior point methods*) of working with inequality constraints $h(x) \leq \mathbf{0}$:

feasible x_k