**Quick Introduction to Optimization**

**Haibin Ling**

---

### Optimization Problems

**Optimization**: determining an argument for which a given function has an extreme value on a given domain.

\[
\min_x f(x) \quad \text{subject to} \quad \begin{cases}
  g(x) = 0 \\
  h(x) \leq 0
\end{cases}
\]

\[
f : \mathbb{R}^n \to \mathbb{R}
\]

\[
g : \mathbb{R}^n \to \mathbb{R}^m
\]

\[
h : \mathbb{R}^n \to \mathbb{R}^p
\]

---

### Example

- **Minimum surface areas of a cylinder given its volume.**

\[
\min f(x_1, x_2) = 2\pi x_1(x_1 + x_2)
\]

subject to

\[
g(x_1, x_2) = \pi x_1^2 x_2 - V = 0
\]

---

### Examples in Practice

- **Support vector machine**
  - Maximize classifier margin
  - Subject to constraints from training data

- **Point cloud matching**
  - Minimize total matching cost
  - Subject to neighborhood constraint

- **Surface warping**
  - Minimize bending energy
  - Subject to smooth constraint (regularization)

- **Bayesian inference**
  - Maximizing posterior probability
  - Energy minimization (MRF, ...)

---

### Unconstrained Optimality Conditions

- **Unconstrained problem**

\[
\min f(x)
\]

- **Derivatives**

\[
\begin{pmatrix}
  \frac{\partial f(x)}{\partial x_1} \\
  \frac{\partial f(x)}{\partial x_2} \\
  \vdots \\
  \frac{\partial f(x)}{\partial x_n}
\end{pmatrix}
\]

\[
\begin{pmatrix}
  \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\
  \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\
  \vdots & \vdots & \ddots & \vdots \\
  \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2}
\end{pmatrix}
\]

---
Unconstrained Optimality Conditions

- Taylor expansion (1st order)
  \[ f(x + s) = f(x) + \nabla f(x) s + \frac{1}{2} s^T H(x) s \]

- First-order necessary condition
  \[ \nabla f(x^*) = 0 \]
  \[ f(x^*) \text{ reaches local minimum at } x^* \Rightarrow \nabla f(x^*) = 0 \]

- Taylor expansion (2nd order)
  \[ f(x^* + s) = f(x^*) + \nabla f(x^*) s + \frac{1}{2} s^T H(x^*) s \]

- Second-order sufficient condition
  \[ s^T H(x^*) s > 0 \quad \forall s \in \mathbb{R}^n \]

  A.k.a. \( H \) positive definite

Summary, for
\[ \min f(x) \]

- Given a critical point \( x^* \), i.e., \( \nabla f(x^*) = 0 \)
  We have
  \[ H \) positive definite \( \Rightarrow x^* \text{ is a minimum of } f \]
  \[ H \) negative definite \( \Rightarrow x^* \text{ is a maximum of } f \]
  \[ H \) indefinite \( \Rightarrow x^* \text{ is a saddle of } f \]
  \[ H \) singular \( \Rightarrow x^* \text{ is pathological} \]

Constrained Optimality Conditions

- Equality constraints
  \[ \min f(x) \quad \text{subject to } \quad g(x) = 0 \]
  From the first-order necessary condition, we have for minimum \( x^* \)
  \[ \nabla f(x^*) s = 0 \]
  Constrain on \( s \) feasible direction
  \[ -\nabla f(x^*) \text{ lies in the space spanned by the constraint normals:} \]
  \[ -\nabla f(x^*) = J_f(x^*)^T \sum \lambda_i g_i(x) \]

- Lagrange function
  \[ \min f(x) \quad \text{subject to } \quad g(x) = 0 \]
  \[ L(x, \lambda) = f(x) + \lambda^T g(x) \]
  Now the problem becomes "unconstrained"
  Similar but a bit more complicated solution

Practice
\[ \min f(x_1, x_2) = 2x_1 x_2 + x_2 \]
subject to
\[ g(x_1, x_2) = x_1^2 + x_2^2 - 4 = 0 \]
\[ L(x_1, x_2, \lambda) = f(x_1, x_2) + \lambda g(x_1, x_2) = 2x_1 x_2 + \lambda(x_1^2 + x_2^2) \]

Solutions for Nonconstrained Problems
No Constraints – 1D

- Some straightforward methods
  - Golden Section Search
  - Successive Parabolic Interpolation

- Newton’s Method
  - Second-order Taylor
    \[ f(x+s) = f(x) + f'(x)s + \frac{1}{2} f''(x)s^2 \]
  - Minimum in \( s \)
    \[ s = -f'(x) / f''(x) \]
  - Iterative algorithm
    \[ s_{i+1} = s_i - f'(x_i) / f''(x_i) \]

No Constraints – High Dimensional

- Direct search
  - Good for nonsmooth objective functions
  - Slow for large dimensional problems

- Steepest Descent
  - Intuition: gradient points to the steepest direction
    \[ x_k = \text{initial guess} \]
    \[ \text{for } i = 0, 1, 2, \ldots \]
    \[ \text{Find gradient } \nabla f(x_k) \]
    \[ \text{Line search } \alpha_i \]
    \[ \text{Choose } \alpha_i \text{ to minimize } f(x_k + \alpha_i s_k) \]
    \[ x_{k+1} = x_k + \alpha_i s_k \]
    \[ \text{end} \]

No Constraints – High Dimensional

- Problem of steepest descent
  - Slow convergence rate

- Newton’s Method
  - Second-order Taylor
    \[ f(x+s) = f(x) + \nabla f(x) s + \frac{1}{2} s^T H(x) s \]
  - Initial guess
    \[ x_0 = \text{initial guess} \]
  - Solve \( H(x_k) s_k - \nabla f(x_k) \)
  - for \( s_k \)
    \[ s_{i+1} = s_i + h_i \]
  - No line search
  - Variants of Newton’s method
    - Quasi-Newton
    - Scant Updating

Solutions for Nonconstrained Problems

Nonlinear Least Squares

Input: \((t, y), i=1, \ldots, m\)
Target: fit a function \(f(t, x)\), where \(x\) can be viewed as parameters.
Residual: \(r(x) = y_j - f(t_j, x)\)

Optimization: minimize the total residual
\[ \phi(x) = \frac{1}{2} \sum_i (r_i)^2 \]
\[ \nabla \phi(x) = -J^T(x) r(x) \]
\[ H(x) = J^T(x) J(x) + \lambda \Delta I \]
Newton’s method:
\[ J^T(x_k) r(x_k) + \sum_i r_i(x_k) H_{ij}(x_k) s_{ij} = -J^T(x_k) r(x_k) \]
Hessian: \(H\) Gaussian: \(\nabla \phi\)

Problem: slow due to Hessian

Nonlinear Least Squares

Newton’s method:
\[ J^T(x_k) r(x_k) + \sum_i r_i(x_k) H_{ij}(x_k) s_{ij} = -J^T(x_k) r(x_k) \]

Observation: \(r_i\) are usually small for a good fitting
\[ J^T(x_k) r(x_k) \]

\nu Gauss-Newton: \(J(x_k) s_k = -r(x_k)\)

Drawback: when some \(r_i\) are not small
Levenberg-Marquardt method:
\[ J^T(x_k) r(x_k) + \lambda \alpha \Delta I s_{ij} = -J^T(x_k) r(x_k) \]
regularization
Solutions for Constrained Problems

Sequential Quadratic Programming

Equality-constrained problems: \( \min f(x) \text{ subject to } g(x) = 0 \)
Lagrange function: \( L(x, \lambda) = f(x) + \lambda^T g(x) \)

\( \Rightarrow \) Solve the following equations

\[
\nabla L(x, \lambda) = \left[ \nabla f(x) + \lambda^T \nabla g(x) \right] = 0
\]

Applying Newton’s method and then after some derivation

\[
\min \frac{1}{2} x^T B(x, \lambda) x + x^T \left[ \nabla f(x) + \lambda^T \nabla g(x) \right]
\text{ subject to } \lambda^T g(x) = 0
\]

where \( B(x, \lambda) = \sum \lambda_i H_i(x) \)

This is done iteratively \( \Rightarrow \) Sequential quadratic programming

Penalty Methods

Equality-constrained problems: \( \min f(x) \text{ subject to } g(x) = 0 \)
Construct an unconstrained problem: \( \min \phi(x) = f(x) + \frac{1}{2} \rho g(x)^T g(x) \)

\( \lim_{\rho \to \infty} x^* = x^* \) solution

Penalize for non-zero \( g(x) \) \( \Rightarrow \) Penalty function method.

Drawbacks: becomes unstable when \( \rho \to \infty \).

Barrier Methods

Motivation: keep intermediate points feasible.

Inequality-constrained problems: \( \min f(x) \text{ subject to } h(x) \geq 0 \)
Construct barrier functions:
Inverse barrier: \( \phi(x) = f(x) - \mu \sum \log(h_i(x)) \)
Logarithmic barrier: \( \phi(x) = f(x) - \mu \sum \log(-h_i(x)) \)

\( \lim_{\mu \to 0} x^* = x^* \) solution

Solves a sequence of unconstrained problems, with decreasing \( \mu \).

Another intuition: from the interior of the feasible set to search for the minimum solution \( \Rightarrow \) interior-point methods

Special Optimization Problems

Linear Programming

\[
\text{min } c^T x + d \\
\text{subject to } Gx \leq h \\
Ax = b
\]

Simplex method: search in the corners of feasible solution polyhedron.
Practically very efficient.
Theoretically can be exponentially expensive.

Interior method: search start from interior.
Proved to have polynomial efficiency in worst case.
Quadratic Programming

\[ \min \frac{1}{2} x^T P x + q^T x + r \]
subject to \( G x \leq h \)
\( A x = b \)

Objective is quadratic, which is convex.
More general case, convex programming.