Numerical optimization : theory and applications

Ammar Mian
Associate professor, LISTIC, Université Savoie Mont Blanc

LISTIC

UNIVERSITÉ
SAVOIE
MONT BLANC

Outline

1. Unconstrained Optimization - Basics
2. What is a Solution?
3. Taylor’s Theorem and Optimality Conditions
4. Optimization Algorithms
    - Steepest Descent Method
    - Newton Method
5. Looking Ahead

1. Unconstrained Optimization - Basics
2. What is a Solution?
3. Taylor's Theorem and Optimality Conditions
4. Optimization Algorithms
    - Steepest Descent Method
    - Newton Method
5. Looking Ahead

Problem Formulation

Unconstrained Optimization Problem

We aim to solve:

$$\operatorname*{argmin}_{x \in \mathbb{R}^d} f(x)$$

where:

- $x \in \mathbb{R}^d$ is the optimization variable
- $f : \mathcal{D}_f \mapsto \mathbb{R}$ is the objective function
- No constraints on the admissible solutions

Goal: Characterize the nature of solutions under this setup.

Local vs Global Minima

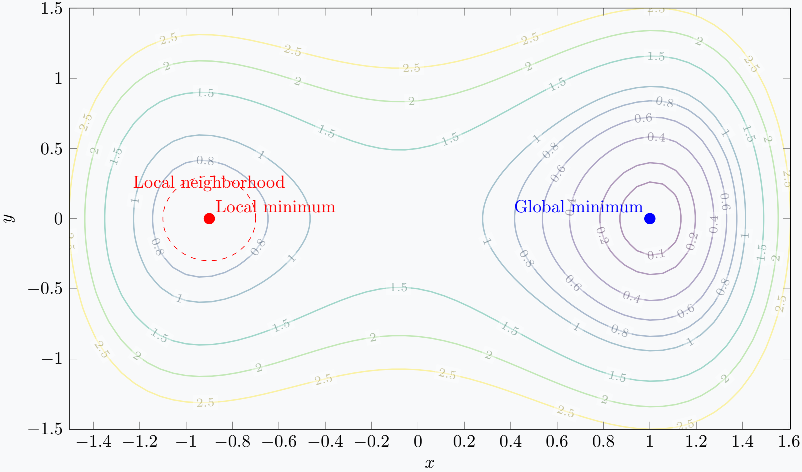

Figure: Local and global minima can coexist in the same function.

Global Minimizer

Definition (Global minimizer)

A point $x^\star$ is a **global minimizer** if

$$f(x^\star) \leq f(x)$$

where x ranges over all of $\mathbb{R}^d$ (or at least over the domain of interest).

- Global minimizers can be difficult to find
- Our knowledge of f is usually only local
- Algorithms typically don't visit many points
- Cannot guarantee finding global minimum in general

Local Minimizer

Definition (Local minimizer)

A point $x^\star$ is a **local minimizer** if

$$\exists r > 0, \quad f(x^\star) \leq f(x), \quad \forall x \in \mathcal{B}(x^\star, r)$$

Types of Local Minimizers

- **Weak local minimizer:** satisfies the definition above
- **Strict local minimizer:** when $f(x^\star) < f(x)$ for all $x \neq x^\star$ in the neighborhood

Taylor's Theorem

Theorem (Taylor's theorem)

Suppose $f : \mathbb{R}^d \mapsto \mathbb{R}$ *is continuously differentiable and* $\mathrm{p} \in \mathbb{R}^d$. *Then:*

$$f(\mathrm{x}+\mathrm{p}) = f(\mathrm{x}) + \nabla f(\mathrm{x}+t\mathrm{p})^T \mathrm{p}$$

for some $t \in [0,1]$.

Moreover, if f *is twice continuously differentiable:*

$$f(\mathrm{x}+\mathrm{p}) = f(\mathrm{x}) + \nabla f(\mathrm{x})^T \mathrm{p} + \frac{1}{2}\mathrm{p}^T \nabla^2 f(\mathrm{x}+t\mathrm{p})\mathrm{p}$$

for some $t \in [0,1]$.

Taylor's Approximation

Theorem (Taylor's approximation)

First order approximation:

$$f(\mathrm{x}+\mathrm{p}) = f(\mathrm{x}) + \nabla f(\mathrm{x})^T \mathrm{p} + o(\|\mathrm{p}\|)$$

Second-order approximation:

$$f(\mathrm{x}+\mathrm{p}) = f(\mathrm{x}) + \nabla f(\mathrm{x})^T \mathrm{p} + \frac{1}{2}\mathrm{p}^T \nabla^2 f(\mathrm{x})\mathrm{p} + o(\|\mathrm{p}\|^2)$$

where $o(\|\mathrm{p}\|)$ and $o(\|\mathrm{p}\|^2)$ represent terms that grow slower than $\|\mathrm{p}\|$ and $\|\mathrm{p}\|^2$ respectively as $\|\mathrm{p}\| \to 0$.

First-Order Necessary Conditions

Theorem (First-order necessary conditions)

If $x^\star$ is a local minimizer, and f is continuously differentiable in a neighborhood of $x^\star$, then

$$\nabla f(x^\star) = 0$$

Stationary Points

We call any point x such that $\nabla f(x) = 0$ a **stationary point**.

Matrix Definiteness

Definitions

A matrix B is:

- **Positive definite** if $p^T Bp > 0$ for all $p \neq 0$
- **Positive semidefinite** if $p^T Bp \geq 0$ for all p

Second-Order Necessary Conditions

Theorem (Second-order necessary conditions)

If $x^\star$ is a local minimizer of f and $\nabla^2 f$ is continuous in an open neighborhood of $x^\star$, then:

- $\nabla f(x^\star) = 0$
- $\nabla^2 f(x^\star)$ *is positive semidefinite*

Second-Order Sufficient Conditions

Theorem (Second-Order Sufficient Conditions)

Suppose that $\nabla^2 f$ is continuous in an open neighborhood of $x^\star$ and that:

- $\nabla f(x^\star) = 0$
- $\nabla^2 f(x^\star)$ *is positive definite*

Then $x^\star$ is a strict local minimizer of f.

Note

These sufficient conditions are not necessary. Example: $f(x) = x^4$ at $x^\star = 0$ is a strict local minimizer but the Hessian vanishes.

1. Unconstrained Optimization - Basics
2. What is a Solution?
3. Taylor's Theorem and Optimality Conditions
4. Optimization Algorithms
   - Steepest Descent Method
   - Newton Method
5. Looking Ahead

The Need for Algorithms

Why do we need algorithms?

- We know that $\nabla f(x^\star) = 0$ characterizes local minima
- But we don't always have the luxury to solve $\nabla f(x) = 0$ analytically
- Computing and checking Hessian conditions can be expensive

Algorithmic Approach

Design iterative algorithms that update x until convergence to a local minimizer:

- **Gradient-based methods:** use only gradient information
- **Newton methods:** use Hessian to accelerate convergence
- **Quasi-Newton methods:** approximate Hessian for balance

Steepest Descent Algorithm

Algorithm

Choose initial point x_0 and compute:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

where α_k are scalar values called **step-size** (or learning rate).

Intuition

- Like walking down a mountain in fog
- Feel the slope and step in steepest descent direction
- $-\nabla f(x_k)$ points in direction of steepest decrease
- Most aggressive local progress toward reducing function value

Steepest Descent - Successful Optimization

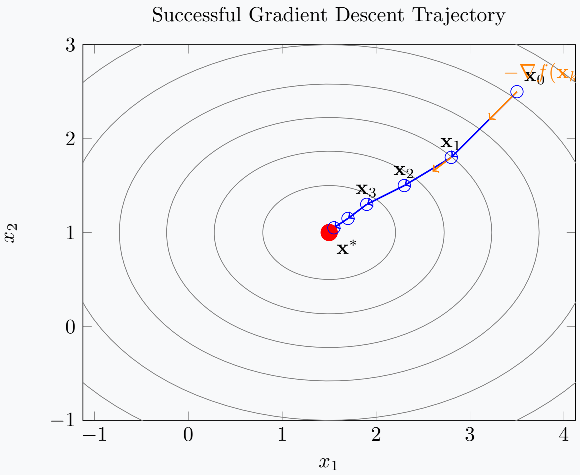

Figure: Successful optimization with steepest descent.

Step Size Trade-offs

Step Size α_k Considerations

The choice of step size involves a fundamental trade-off:

- **Too small:** Painfully slow progress
- **Too large:** Might overshoot or start climbing uphill

Steepest Descent - Zigzag Problem

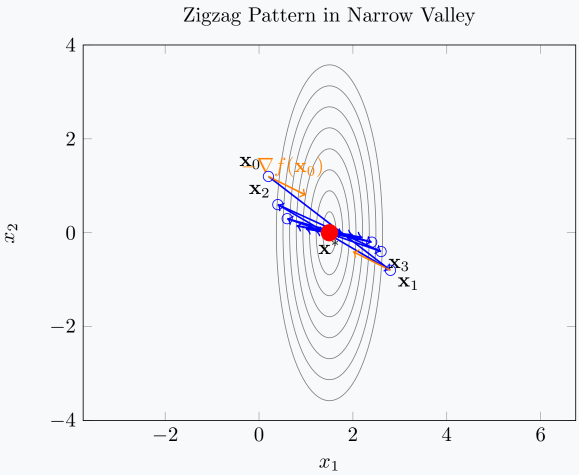

Figure: Zigzag behavior of steepest descent in narrow valleys.

Why Steepest Descent Can Struggle

Zigzag Behavior

- Occurs in narrow valley-like functions (large condition number)
- Algorithm bounces between valley walls instead of walking down
- Steepest direction points toward walls, not down the valley
- Fundamentally myopic: only considers immediate local slope

Convergence Properties

- Linear convergence under reasonable conditions
- Error decreases by constant factor each iteration
- Can be frustratingly slow for poorly conditioned problems

1. Unconstrained Optimization - Basics
2. What is a Solution?
3. Taylor’s Theorem and Optimality Conditions
4. Optimization Algorithms
    - Steepest Descent Method
    - Newton Method
5. Looking Ahead

Newton Method - Motivation

Key Insight

- Steepest descent: navigating with only immediate slope
- Newton method: having detailed topographic map
- Incorporates curvature information (how slope changes)
- Uses second-order Taylor approximation

Strategy

Instead of minimizing f directly, minimize simpler quadratic approximation:

$$f(x_k + p) \approx f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T \nabla^2 f(x_k) p$$

Newton Method - Algorithm

Derivation

Setting gradient of quadratic approximation to zero:

$$\nabla f(x_k) + \nabla^2 f(x_k)p = 0$$

Solving for Newton step:

$$p_k = -[\nabla^2 f(x_k)]^{-1} \nabla f(x_k)$$

Newton Iteration

$$x_{k+1} = x_k - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k)$$

Newton Method - Optimization Step

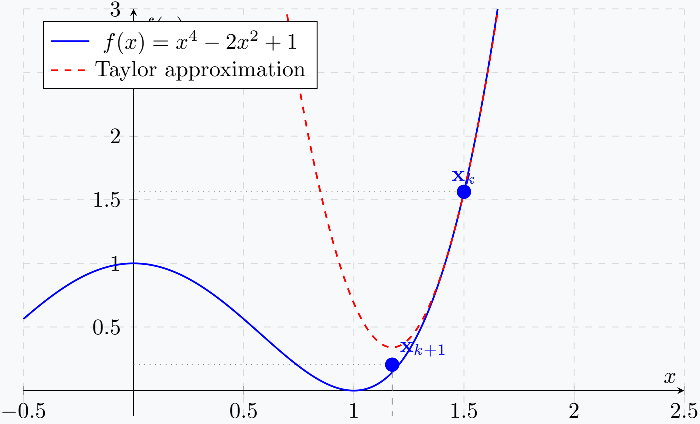

Figure: Newton method optimization step.

Newton Method - Geometric Insight

Hessian Information

- $\nabla^2 f(x_k)$ encodes how gradient changes in different directions
- Recognizes elongated valley shapes
- Takes larger steps along valley floor, smaller steps perpendicular
- Eliminates zigzag behavior of steepest descent

Special Property

For quadratic functions: Newton method finds exact minimum in single step, regardless of conditioning!

Newton Method - Convergence

Quadratic Convergence

Near solution satisfying second-order sufficient conditions:

- Number of correct digits roughly doubles each iteration
- If 1 correct digit � next iteration gives 2 � then 4 � then 8
- Incredibly efficient for high-precision optimization
- Forms backbone of many sophisticated algorithms

Comparison

- **Linear convergence:** 1 digit � 3 iterations � 2 digits
- **Quadratic convergence:** 1 digit � 1 iteration � 2 digits

Newton Method - Computational Cost

The Price of Power

- Must compute Hessian matrix: $d(d+1)/2$ second derivatives
- Must solve linear system: $\nabla^2 f(\mathrm{x}_k)\mathrm{p}_k = -\nabla f(\mathrm{x}_k)$
- Requires $\sim d^3/3$ arithmetic operations per iteration
- Becomes prohibitive as dimension d grows

Newton Method - Potential Failures

When Newton's Method Can Fail

- Hessian might not be positive definite away from minimum
- Quadratic model might have maximum or saddle point
- Newton step might point in wrong direction
- Poor quadratic approximation far from minimum
- Steps might increase function value

1. Unconstrained Optimization - Basics

2. What is a Solution?

3. Taylor’s Theorem and Optimality Conditions

4. Optimization Algorithms
    - Steepest Descent Method
    - Newton Method

5. Looking Ahead

Bridge Between Methods

Motivating Questions

- Can we capture Newton's geometric insight without full computational burden?
- Can we ensure global reliability while achieving faster convergence?

Advanced Methods

- **Quasi-Newton methods (BFGS):** Approximate Hessian using gradients, superlinear convergence
- **Trust region methods:** Systematic progress guarantees
- **Line search strategies:** Reliable step size selection