# Applications of Linear Algebra in Optimization Algorithms

Nino Wang

October 31, 2025

# Outline

# Introduction to Optimization

- Optimization: Finding the best solution from a set of possible solutions
- Mathematical formulation:

$$\min_{x \in \mathbb{R}^n} f(x) \tag{1}$$

$$\text{subject to } g_i(x) \leq 0, \quad i = 1, \ldots, m \tag{2}$$

$$h_j(x) = 0, \quad j = 1, \ldots, p \tag{3}$$

- Classes of optimization problems:
  - Unconstrained vs. Constrained
  - Linear vs. Nonlinear
  - Convex vs. Non-convex
- Linear algebra provides the mathematical foundation for many optimization algorithms

# Convexity and Linear Algebra

- Convex set: For any two points $x, y$ in the set, the line segment between them is also in the set

$$\theta x + (1-\theta)y \in \text{set}, \quad \forall \theta \in [0,1] \tag{4}$$

- Convex function: Second-order condition uses linear algebra

$$f \text{ is convex} \iff \nabla^2 f(x) \succeq 0 \quad \forall x \tag{5}$$

- Hessian matrix (second derivative matrix):

$$H_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j} \tag{6}$$

- Positive definiteness: For all $z \neq 0$, $z^T H z > 0$

# Proving Norm Convexity Using Matrix Calculus

- Vector norms are essential in optimization (objective functions, constraints)
- Claim: Any norm $\|x\|$ is a convex function
- Proof approach: Use second-order conditions with matrix calculus
- For the $\ell_2$ norm $\|x\|_2 = \sqrt{x^T x}$, computing the Hessian:

$$f(x) = \|x\|_2 = \sqrt{x^T x} \tag{7}$$

$$\nabla f(x) = \frac{x}{\|x\|_2} \tag{8}$$

$$\nabla^2 f(x) = \frac{I}{\|x\|_2} - \frac{xx^T}{\|x\|_2^3} \tag{9}$$

- For any vector $z \perp x$: $z^T(\nabla^2 f(x))z = \frac{\|z\|_2^2}{\|x\|_2} > 0$
- For vectors parallel to $x$: $\lambda x^T(\nabla^2 f(x))\lambda x = 0$
- Therefore, $\nabla^2 f(x)$ is positive semidefinite $\Rightarrow \|x\|_2$ is convex
- General norms: use the fact that a function is convex if and only if its restriction to any line is convex

# Example: Checking Convexity

Consider the quadratic function:

$$f(x) = x^T A x + b^T x + c \tag{10}$$

where $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, and $c \in \mathbb{R}$.

Computing the Hessian:

$$\nabla f(x) = 2Ax + b \tag{11}$$

$$\nabla^2 f(x) = 2A \tag{12}$$

Therefore, $f$ is convex if and only if $A$ is positive semidefinite.

Checking positive definiteness:

- Compute eigenvalues of $A$: all must be non-negative
- Compute all principal minors: all must be non-negative
- Cholesky decomposition: must exist ($A = LL^T$)

# Gradient Descent Method

- Key idea: Move in the direction of steepest descent
- Update rule:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k) \tag{13}$$

  where $\alpha_k$ is the step size
- Linear algebra connection: The gradient direction is orthogonal to the level sets of $f$

$$\langle \nabla f(x), y - x \rangle > 0 \iff f(y) > f(x) \text{ for } y \text{ close to } x \tag{14}$$

- This uses the inner product space structure of $\mathbb{R}^n$
- Convergence rate depends on the condition number of the Hessian matrix

# Newton's Method

- Uses second-order information (Hessian matrix)
- Quadratic approximation at current point:

$$f(x) \approx f(x_k) + \nabla f(x_k)^T (x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k) \tag{15}$$

- Update rule:

$$x_{k+1} = x_k - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k) \tag{16}$$

- Linear algebra operations:
  - Matrix inversion (or solving linear system)
  - Matrix-vector multiplication
- Faster convergence but more computationally expensive per iteration

# Linear Programming

- Standard form:

$$\min_{x} \quad c^T x \tag{17}$$
$$\text{subject to} \quad Ax = b \tag{18}$$
$$x \geq 0 \tag{19}$$

  where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$
- Simplex method: Linear algebra at each step
  - Basis matrix $B$ (submatrix of $A$)
  - Basic feasible solution: $x_B = B^{-1}b$
  - Reduced costs: $\bar{c}_j = c_j - c_B^T B^{-1} A_j$
  - Pivot operations: Row operations (Gaussian elimination)

# Simplex Method: Linear Algebra Perspective

- Each iteration involves:
  - Computing $B^{-1}$: Matrix inversion
  - Computing $B^{-1}b$ and $B^{-1}A_j$: Matrix-vector products
  - Pivot operation: Elementary row operations
- Geometrically: Moving from one vertex of the feasible region to an adjacent vertex
- Linear independence of constraint vectors determines the dimension of the feasible region
- Interpretation of basic and non-basic variables:
  - Basic variables: Correspond to linearly independent columns of $A$
  - Non-basic variables: Set to zero to solve the system $Ax = b$

# Gradient-Based Methods in Inner Product Spaces

- Gradient direction uses inner product structure

$$\nabla f(x) = \arg \max_{v:\|v\|=1} D_v f(x) \tag{20}$$

- Conjugate gradient method: Builds orthogonal directions

$$p_{k+1} = -\nabla f(x_{k+1}) + \beta_k p_k \tag{21}$$

$$\beta_k = \frac{\nabla f(x_{k+1})^T \nabla f(x_{k+1})}{\nabla f(x_k)^T \nabla f(x_k)} \tag{22}$$

- Weighted inner products: Scaling the space

$$\langle x, y \rangle_M = x^T M y \tag{23}$$

  where $M$ is positive definite

- Preconditioned methods: Change the inner product to improve convergence

# Conclusion and Advanced Topics

- Linear algebra is fundamental to optimization algorithms:
  - Matrix analysis for convexity conditions
  - Vector spaces and inner products for gradient methods
  - Matrix operations in iterative methods
  - Linear transformations in constrained optimization
- Advanced topics:
  - Singular Value Decomposition (SVD) for low-rank approximation
  - QR decomposition for least squares problems
  - Eigenvalue problems in semidefinite programming
  - Krylov subspace methods for large-scale problems
- Computational considerations:
  - Exploiting matrix structure (sparsity, symmetry)
  - Numerical stability of linear algebra operations
  - Parallel implementation of matrix operations