Statistics and Machine Learning
Fall, 2005

Lecture 2: Optimization
You Have Learned (Unconstrained) Optimization in Your High School

Let \( f(x) = ax^2 + bx + c, \ a \neq 0, \ x^* = -\frac{b}{2a} \)

Case I: \( f''(x^*) = a > 0 \Rightarrow x^* \in \arg\min_{x \in \mathbb{R}} f(x) \)

Case II: \( f''(x^*) = a < 0 \Rightarrow x^* \in \arg\max_{x \in \mathbb{R}} f(x) \)

For minimization problem (Case I),

\( f'(x^*) = 0 \) is called the first order optimality condition

\( f''(x^*) = a > 0 \) is the second order optimality condition
Optimization Examples in this Book

On p.62, Maximum likelihood estimation
On p.68, Maximum a Posteriori estimation
On p.74, Least squares estimates
On p.207, Gradient descent method
On p.246, Backpropagation
Gradient and Hessian

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable function. The gradient of function $f$ at a point $x \in \mathbb{R}^n$ is defined as

$$\nabla f(x) = \left[ \frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \ldots, \frac{\partial f(x)}{\partial x_n} \right] \in \mathbb{R}^n$$

If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a twice differentiable function. The Hessian matrix of $f$ at a point $x \in \mathbb{R}^n$ is defined as

$$\nabla^2 f(x) = \begin{bmatrix}
\frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\
\frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\
\cdots & \cdots & \cdots & \cdots \\
\frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2}
\end{bmatrix} \in \mathbb{R}^{n \times n}$$
Example of Gradient and Hessian

\[ f(x) = x_1^2 + x_2^2 - 2x_1 + 4x_2 \]
\[ = \frac{1}{2} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} -2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \]

\[ \nabla f(x) = [2x_1 - 2, 2x_2 + 4] \quad \nabla^2 f(x) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \]

By letting \( \nabla f(x) = 0 \), we have

\[ x^* = \begin{bmatrix} 1 \\ -2 \end{bmatrix} \in \arg \min_{x \in \mathbb{R}^2} f(x) \]
Quadratic Functions (Standard Form)

\[ f(x) = \frac{1}{2} x^T H x + p^T x \]

Let \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) and \( f(x) = \frac{1}{2} x^T H x + p^T x \)

where \( H \in \mathbb{R}^{n \times n} \) is a symmetric matrix and \( p \in \mathbb{R}^n \)

then
\[
\nabla f(x) = Hx + p
\]
\[
\nabla^2 f(x) = H \quad \text{(Hessian)}
\]

Note: If \( H \) is positive definite, then \( x^* = -H^{-1}p \)

is the unique solution of \( \min f(x) \)
Least-squares Problem

\[ \min_{x \in \mathbb{R}^{n}} \| Ax - b \|_2^2, \quad A \in \mathbb{R}^{m \times n}, \quad b \in \mathbb{R}^m \]

\[ f(x) = (Ax - b)^T(Ax - b) \]
\[ = x^T A^T Ax - 2b^T Ax + b^T b \]

\[ \nabla f(x) = 2A^T Ax - 2A^T b \]

\[ \nabla^2 f(x) = 2A^T A \]

\[ x^* = (A^T A)^{-1} A^T b \in \arg \min_{x \in \mathbb{R}^n} \| Ax - b \|_2^2 \]

if \( A^T A \) is nonsingular matrix \( \Rightarrow \) P.D.

Note: \( x^* \) is an analytical solution
How to Solve an Unconstrained MP

- Get an initial point and iteratively decrease the objective function value
- Stop once the stopping criteria are satisfied
- Steep decent might not be a good choice
- Newton’s method is highly recommended
  - Local and quadratic convergent algorithm
  - Need to choose a good step size to guarantee global convergence
The First Order Taylor Expansion

Let \( f : \mathbb{R}^n \to \mathbb{R} \) be a differentiable function

\[
f(x + d) = f(x) + \nabla f(x) \cdot d + \alpha(x, d) \|d\|
\]

where \( \lim_{d \to 0} \alpha(x, d) = 0 \)

If \( \nabla f(x) d < 0 \) and \( d \) is small enough

then \( f(x + d) < f(x) \).

We call \( d \) is a descent direction.
Steep Descent with Exact Line Search

Start with any $x^0 \in \mathbb{R}^n$. Having $x^i$, stop if $\nabla f(x^i) = 0$

Else compute $x^{i+1}$ as follows:

(i) Steep descent direction: $d^i = - \nabla f(x^i)$

(ii) Exact line search: Choose a stepsize $\lambda \in \mathbb{R}$ such that

$$\frac{d f(x^i + \lambda d^i)}{d \lambda} = f'(x^i + \lambda d^i) = 0$$

(iii) Updating: $x^{i+1} = x^i + \lambda d^i$
MATLAB Code for Steep Descent with Exact Line Search (Quadratic Function Only)

```matlab
function [x, f_value, iter] = grdlines(Q, p, x0, esp)

% % min 0.5*x'Q*x+p'x
% % Solving unconstrained minimization via
% % steep descent with exact line search
% 
```
flag = 1;
iter = 0;
while flag > esp
    grad = Q*x0+p;
    temp1 = grad'*grad;
    if temp1 < 10^-12
        flag = esp
    else
        stepsize = temp1/(grad'*Q*grad);
        x1 = x0 - stepsize*grad;
        flag = norm(x1-x0);
        x0 = x1;
    end;
    iter = iter+1;
end;
x = x0;
f_value = 0.5*x'*Q*x+p'*x;
The Key Idea of Newton’s Method

Let \( f : R^n \rightarrow R \) be a twice differentiable function

\[
f(x + d) = f(x) + \nabla f(x) \cdot d + \frac{1}{2} x^T \nabla^2 f(x)x + \beta(x, d) \|d\|
\]

where \( \lim_{d \rightarrow 0} \beta(x, d) = 0 \)

At \( i^{th} \) iteration, use a quadratic function to approximate

\[
f(x) \approx f(x^i) + \nabla f(x^i)(x - x^i) + \frac{1}{2}(x - x^i)^T \nabla^2 f(x^i)(x - x^i)
\]

\[x^{i+1} = \arg \min f(x)\]
Newton’s Method

Start with $x^0 \in \mathbb{R}^n$. Having $x^i$, stop if $\nabla f(x^i) = 0$

Else compute $x^{i+1}$ as follows:

(i) Newton direction: $\nabla^2 f(x^i) d^i = -\nabla f(x^i)$.

Have to solve a system of linear equations here!

(ii) Updating: $x^{i+1} = x^i + d^i$

➢ Converge only when $x^0$ is close to $x^*$ enough.
\[ f(x) = - \frac{1}{6} x^6 + \frac{1}{4} x^4 + 2 x^2 \]
\[ g(x) = f(x^i) + f'(x^i)(x - x^i) + \frac{1}{2} f''(x^i)(x - x^i) \]

It can not converge to the optimal solution.
Constrained Optimization Problem

Problem setting: Given functions $f, g_i, i = 1, \ldots, k$ and $h_j, j = 1, \ldots, m$, defined on a domain $\Omega \subseteq \mathbb{R}^n$,

$$\min_{x \in \Omega} f(x)$$

subject to $g_i(x) \leq 0, \forall i$

$h_j(x) = 0, \forall j$

where $f(x)$ is called the objective function and $g(x) \leq 0, h(x) = 0$ are called constraints.
Example

$$\min \ f(x) = 2x_1^2 + x_2^2 + 3x_3^2$$

s.t.  $$2x_1 - 3x_2 + 4x_3 = 49$$

<sol>

$$L(x, \alpha) = f(x) + \beta(2x_1 - 3x_2 + 4x_3 - 49) \quad \beta \in \mathbb{R}$$

$$\frac{\partial}{\partial x_1} L(x, \beta) = 0 \quad \Rightarrow \quad 4x_1 + 2\beta = 0$$

$$\frac{\partial}{\partial x_2} L(x, \beta) = 0 \quad \Rightarrow \quad 2x_2 - 3\beta = 0$$

$$\frac{\partial}{\partial x_3} L(x, \beta) = 0 \quad \Rightarrow \quad 6x_3 + 4\beta = 0$$

$$2x_1 - 3x_2 + 4x_3 - 49 = 0$$

$$\Rightarrow \ x_1 = 3, \ x_2 = -9, \ x_3 = -4$$
\[
\min_{x \in \mathbb{R}^2} \quad x_1^2 + x_2^2 \\
\quad x_1 + x_2 \leq 4 \\
\quad -x_1 - x_2 \leq -2 \\
\quad x_1, \quad x_2 \geq 0
\]

\[
\nabla f(x) = [2x_1, 2x_2]
\]

\[
\nabla f(x^*) = [2, 2]
\]
Definitions and Notation

- **Feasible region:**

  \[ \mathcal{F} = \{ x \in \Omega | \ g(x) \leq 0, \ h(x) = 0 \} \]

  where \( g(x) = \begin{bmatrix} g_1(x) \\ \vdots \\ g_k(x) \end{bmatrix} \) and \( h(x) = \begin{bmatrix} h_1(x) \\ \vdots \\ h_m(x) \end{bmatrix} \)

- A solution of the optimization problem is a point \( x^* \in \mathcal{F} \) such that \( \nexists x \in \mathcal{F} \) for which \( f(x) < f(x^*) \) and \( x^* \) is called a global minimum.
Definitions and Notation

- A point $\bar{x} \in \mathcal{F}$ is called a local minimum of the optimization problem if there exists $\varepsilon > 0$ such that
  $$f(x) \geq f(\bar{x}), \quad \forall x \in \mathcal{F} \text{ and } \|x - \bar{x}\| < \varepsilon$$

- At the solution $x^*$, an inequality constraint $g_i(x)$ is said to be active if $g_i(x^*) = 0$, otherwise it is called an inactive constraint.

- $g_i(x) \leq 0 \iff g_i(x) + \xi_i = 0, \quad \xi_i \geq 0$ where $\xi_i$ is called the slack variable
Definitions and Notation

- Remove an inactive constraint in an optimization problem will **NOT** affect the optimal solution
  - Very useful feature in SVM
- If $\mathcal{F} = \mathbb{R}^n$ then the problem is called **unconstrained** minimization problem
  - Least square problem is in this category
  - SSVM formulation is in this category
  - Difficult to find the global minimum without convexity assumption
The Most Important Concepts in Optimization (minimization)

◆ A point is said to be an *optimal solution* of a unconstrained minimization if there exists no decent direction \( \nabla f(x^*) = 0 \).

◆ A point is said to be an optimal solution of a constrained minimization if there exists no feasible decent direction \( \nabla f(x^*) = 0 \) \( \iff \) KKT conditions.

➢ There might exist decent direction but move along this direction will leave out the feasible region.
Minimum Principle

Let \( f : \mathbb{R}^n \to \mathbb{R} \) be a convex and differentiable function and \( \mathcal{F} \subseteq \mathbb{R}^n \) be the feasible region.

\[
x^* \in \arg \min_{x \in \mathcal{F}} f(x) \iff \nabla f(x^*)(x - x^*) \geq 0 \quad \forall x \in \mathcal{F}
\]

Example:

\[
\min (x - 1)^2 \quad s.t. \quad a \leq x \leq b
\]
\[
\begin{align*}
\min_{x \in \mathbb{R}^2} & \quad x_1^2 + x_2^2 \\
\text{s.t.} & \quad x_1 + x_2 \leq 4 \\
& \quad -x_1 - x_2 \leq -2 \\
& \quad x_1, x_2 \geq 0
\end{align*}
\]

\[\nabla f(x) = [2x_1, 2x_2]\]

\[\nabla f(x^*) = [2, 2]\]
Linear Programming Problem

An optimization problem in which the objective function and all constraints are linear functions is called a linear programming problem.

\[
\begin{align*}
\text{(LP)} && \min & \quad p^T x \\
\text{s.t.} && A x & \leq b \\
&& C x &= d \\
&& L \leq x \leq U
\end{align*}
\]
**Linear Programming Solver in MATLAB**

**X=LINPROG(f,A,b)** attempts to solve the linear programming problem:

\[
\text{min } f^*x \quad \text{subject to: } \quad A^*x \leq b \\
x
\]

**X=LINPROG(f,A,b,Aeq,beq)** solves the problem above while additionally satisfying the equality constraints \( Aeq^*x = beq \).

**X=LINPROG(f,A,b,Aeq,beq,LB,UB)** defines a set of lower and upper bounds on the design variables, \( X \), so that the solution is in the range \( LB \leq X \leq UB \).

Use empty matrices for \( LB \) and \( UB \) if no bounds exist. Set \( LB(i) = -\infty \) if \( X(i) \) is unbounded below; set \( UB(i) = \infty \) if \( X(i) \) is unbounded above.
Linear Programming Solver in MATLAB

`X=LINPROG(f,A,b,Aeq,beq,LB,UB,X0)` sets the starting point to `X0`. This option is only available with the active-set algorithm. The default interior point algorithm will ignore any non-empty starting point.

You can type “help linprog” in MATLAB to get more information!
\[ L_1 \text{-Approximation: } \min_{x \in \mathbb{R}^n} \| Ax - b \|_1 \]

\[ \| z \|_1 = \sum_{i=1}^{m} |z_i| \]

\[ \min_{x,s} \quad 1^T s \quad \text{Or} \quad \min_{x,s} \quad \sum_{i=1}^{m} s_i \]

s.t. \quad -s \leq Ax - b \leq s \quad \text{s.t.} \quad -s_i \leq A_i x - b_i \leq s_i \quad \forall \ i

\[ \min_{x,s} \quad [0 \cdots 0 1 \cdots 1] \begin{bmatrix} x \\ s \end{bmatrix} \]

s.t. \quad \begin{bmatrix} A & -I \\ -A & -I \end{bmatrix}_{2m \times (n+m)} \begin{bmatrix} x \\ s \end{bmatrix} \leq \begin{bmatrix} b \\ -b \end{bmatrix} \]
Chebyshev Approximation: \[
\min_{x \in \mathbb{R}^n} \| Ax - b \|_\infty
\]

\[
\| z \|_\infty = \max_{1 \leq i \leq m} | z_i |
\]

\[
\begin{align*}
\min_{x, \gamma} & \quad \gamma \\
\text{s.t.} & \quad -1 \gamma \leq Ax - b \leq 1 \gamma
\end{align*}
\]

\[
\begin{align*}
\min_{x, \gamma} & \quad [0 \cdots 0 1] \begin{bmatrix} x \\ \gamma \end{bmatrix} \\
\text{s.t.} & \quad \begin{bmatrix} A & -1 \\ -A & -1 \end{bmatrix}_{2m \times (n+1)} \begin{bmatrix} x \\ \gamma \end{bmatrix} \leq \begin{bmatrix} b \\ -b \end{bmatrix}
\end{align*}
\]
If the objective function is convex quadratic while the constraints are all linear then the problem is called convex quadratic programming problem.

\[(QP) \quad \min \quad \frac{1}{2} x^T Q x + p^T x \]
\[\text{s.t.} \quad A x \leq b \]
\[C x = d \]
\[L \leq x \leq U\]
Quadratic Programming Solver in MATLAB

\( X = \text{QUADPROG}(H,f,A,b) \) attempts to solve the quadratic programming problem:

\[
\begin{align*}
\min & \quad 0.5x^\top Hx + f^\top x \\
\text{subject to:} & \quad A^\top x \leq b \\
x & \geq 0
\end{align*}
\]

\( X = \text{QUADPROG}(H,f,A,b,Aeq,beq) \) solves the problem above while additionally satisfying the equality constraints \( Aeq^\top x = beq \).

\( X = \text{QUADPROG}(H,f,A,b,Aeq,beq,LB,UB) \) defines a set of lower and upper bounds on the design variables, \( X \), so that the solution is in the range \( LB \leq X \leq UB \).

Use empty matrices for \( LB \) and \( UB \) if no bounds exist. Set \( LB(i) = -\text{Inf} \) if \( X(i) \) is unbounded below; set \( UB(i) = \text{Inf} \) if \( X(i) \) is unbounded above.
Quadratic Programming Solver in MATLAB

\[ X = \text{QUADPROG}(H,f,A,b,Aeq,beq,LB,UB,X0) \] sets the starting point to \( X_0 \).

You can type “help quadprog” in MATLAB to get more information!
Standard Support Vector Machine

$$\min_{w, b, \xi_A, \xi_B} C(1^T \xi_A + 1^T \xi_B) + \frac{1}{2} \| w \|_2^2$$

$$(A w + 1 b) + \xi_A \geq 1$$

$$(B w + 1 b) - \xi_B \leq -1$$

$$\xi_A \geq 0, \xi_B \geq 0$$
Farkas’ Lemma

For any matrix \( A \in \mathbb{R}^{m \times n} \) and any vector \( b \in \mathbb{R}^n \), either
\[
Ax \geq 0, \quad b^T x < 0 \text{ has a solution}
\]
or
\[
A^T \alpha = b, \quad \alpha \geq 0 \text{ has a solution}
\]
but never both.
Minimization Problem \textit{vs.} Kuhn-Tucker Stationary-point Problem

\textbf{MP:} \quad \min_{x \in \Omega} f(x) \quad \text{such that} \quad g(x) \leq 0

\textbf{KTSP:} \quad \text{Find } \overline{x} \in \Omega, \; \overline{\alpha} \in \mathbb{R}^m \quad \text{such that}

\nabla f(\overline{x}) + \overline{\alpha}' \nabla g(\overline{x}) = 0

\overline{\alpha}' g(\overline{x}) = 0

g(\overline{x}) \leq 0, \; \overline{\alpha} \geq 0
Lagrangian Function
\[ \mathcal{L}(x, \alpha) = f(x) + \alpha'g(x) \]

Let \( \mathcal{L}(x, \alpha) = f(x) + \alpha'g(x) \) and \( \alpha \geq 0 \)

◆ If \( f(x), g(x) \) are convex then \( \mathcal{L}(x, \alpha) \) is convex.

◆ For a fixed \( \alpha \geq 0 \), if \( \bar{x} \in \arg \min \{ \mathcal{L}(x, \alpha) \mid x \in \mathbb{R}^n \} \)
then
\[ \frac{\partial \mathcal{L}(x, \alpha)}{\partial x} \bigg|_{x=\bar{x}} = \nabla f(\bar{x}) + \alpha' \nabla g(\bar{x}) = 0 \]

◆ Above result is a sufficient condition if \( \mathcal{L}(x, \alpha) \) is convex.
KTSP with Equality Constraints?

(Assume $h(x) = 0$ are linear functions)

$h(x) = 0 \iff h(x) \leq 0$ and $-h(x) \leq 0$

**KTSP:** Find $\bar{x} \in \Omega$, $\bar{\alpha} \in \mathbb{R}^k$, $\bar{\beta}_+, \bar{\beta}_- \in \mathbb{R}^m$ such that

$$\nabla f(\bar{x}) + \bar{\alpha}' \nabla g(\bar{x}) + (\bar{\beta}_+ - \bar{\beta}_-)' \nabla h(\bar{x}) = 0$$

$$\bar{\alpha}' g(\bar{x}) = 0, (\bar{\beta}_+)' h(\bar{x}) = 0, (\bar{\beta}_-)'(-h(\bar{x})) = 0$$

$$g(\bar{x}) \leq 0, h(\bar{x}) = 0$$

$$\bar{\alpha} \succeq 0, \bar{\beta}_+, \bar{\beta}_- \succeq 0$$
KTSP with Equality Constraints

KTSP: Find $\bar{x} \in \Omega$, $\bar{\alpha} \in \mathbb{R}^k$, $\bar{\beta} \in \mathbb{R}^m$ such that

$$\nabla f(\bar{x}) + \bar{\alpha}' \nabla g(\bar{x}) + \bar{\beta}' \nabla h(\bar{x}) = 0$$

$$\bar{\alpha}' g(\bar{x}) = 0, \ g(\bar{x}) \leq 0, \ h(\bar{x}) = 0$$

$$\bar{\alpha} \geq 0$$

\[\text{Let } \bar{\beta} = \bar{\beta}_+ - \bar{\beta}_- \text{ and } \bar{\beta}_+, \bar{\beta}_- \geq 0 \text{ then} \]

$\bar{\beta}$ is free variable
Generalized Lagrangian Function

\[ \mathcal{L}(x, \alpha, \beta) = f(x) + \alpha'g(x) + \beta'h(x) \]

Let \( \mathcal{L}(x, \alpha, \beta) = f(x) + \alpha'g(x) + \beta'h(x) \) and \( \alpha \geq 0 \)

- If \( f(x), g(x) \) are convex and \( h(x) \) is linear then \( \mathcal{L}(x, \alpha, \beta) \) is convex.

- For fixed \( \alpha \geq 0, \beta \), if \( x \in \arg \min \{ \mathcal{L}(x, \alpha, \beta) \mid x \in \mathbb{R}^n \} \) then

\[
\left. \frac{\partial \mathcal{L}(x, \alpha, \beta)}{\partial x} \right|_{x = \bar{x}} = \nabla f(\bar{x}) + \alpha'\nabla g(\bar{x}) + \beta'\nabla h(\bar{x}) = 0
\]

- Above result is a sufficient condition if \( \mathcal{L}(x, \alpha, \beta) \) is convex.
Lagrangian Dual Problem

\[ \max \min \min \mathcal{L}(x, \alpha, \beta) \]
\[ \alpha, \beta \quad x \in \Omega \]
subject to \[ \alpha \geq 0 \]
Lagrangian Dual Problem

$$\max \min_{\alpha, \beta} \mathcal{L}(x, \alpha, \beta)$$
subject to \quad \alpha \geq 0

$$\max_{\alpha, \beta} \theta(\alpha, \beta)$$
subject to \quad \alpha \geq 0

where \quad \theta(\alpha, \beta) = \inf_{x \in \Omega} \mathcal{L}(x, \alpha, \beta)$$
Weak Duality Theorem

Let \( \bar{x} \in \Omega \) be a feasible solution of the primal problem and \( (\alpha, \beta) \) a feasible solution of the dual problem. Then

\[
f(\bar{x}) \geq \theta(\alpha, \beta)
\]

where

\[
\theta(\alpha, \beta) = \inf_{x \in \Omega} L(x, \alpha, \beta) = \inf \{ L(x, \alpha, \beta) | x \in \Omega \}
\]

Corollary: \[
\sup\{ \theta(\alpha, \beta) | \alpha \geq 0 \}
\]

\[
\leq \inf \{ f(x) | g(x) \leq 0, \ h(x) = 0 \}
\]
Weak Duality Theorem

Corollary: If \( f(x^*) = \theta(\alpha^*, \beta^*) \) where \( \alpha^* \geq 0 \)
and \( g(x^*) \leq 0, \ h(x^*) = 0 \), then \( x^* \) and \( (\alpha^*, \beta^*) \)
solve the primal and dual problem respectively. In this case,

\[
0 \leq \alpha \perp g(x) \leq 0
\]
Saddle Point of Lagrangian

Let \( x^* \in \Omega, \alpha^* \geq 0, \beta^* \in \mathbb{R}^m \) satisfying

\[
L(x^*, \alpha, \beta) \leq L(x^*, \alpha^*, \beta^*) \leq L(x, \alpha^*, \beta^*),
\]
\[
\forall \ x \in \Omega, \ \alpha \geq 0. \text{ Then } (x^*, \alpha^*, \beta^*) \text{ is called}
\]

The saddle point of the Lagrangian function
Saddle point of $f(x,y) = x^2 - y^2$
Dual Problem of Linear Program

**Primal LP**

\[
\min_{x \in \mathbb{R}^n} \quad p'x \\
\text{subject to} \quad Ax \geq b, \quad x \geq 0
\]

**Dual LP**

\[
\max_{\alpha \in \mathbb{R}^m} \quad b'\alpha \\
\text{subject to} \quad A'\alpha \leq p, \quad \alpha \geq 0
\]

※ All duality theorems hold and work perfectly!
Lagrangian Function of Primal LP

\[ \mathcal{L}(x, \alpha) = p'x + \alpha'_1(b - Ax) + \alpha'_2(-x) \]

\[
\begin{align*}
\max \ \min_{\alpha_1, \alpha_2 \geq 0} \ \mathcal{L}(x, \alpha_1, \alpha_2) & \iff \\
\max_{\alpha_1, \alpha_2 \geq 0} \ p'x + \alpha'_1(b - Ax) + \alpha'_2(-x) & \iff \\
\ p - A'\alpha_1 - \alpha_2 = 0 \\
(\nabla_x \mathcal{L}(x, \alpha_1, \alpha_2) = 0) 
\end{align*}
\]
Application of LP Duality

**LSQ-Normal Equation Always Has a Solution**

For any matrix $A \in \mathbb{R}^{m \times n}$ and any vector $b \in \mathbb{R}^m$, consider

$$\min_{x \in \mathbb{R}^n} \| Ax - b \|_2^2$$

$$x^* \in \arg\min \{ \| Ax - b \|_2^2 \} \iff A'Ax^* = A'b$$

**Claim:** $A'Ax = A'b$ always has a solution.
Dual Problem of Strictly Convex Quadratic Program

**Primal QP**

\[
\begin{align*}
\min_{x \in \mathbb{R}^n} & \quad \frac{1}{2} x'Qx + p'x \\
\text{subject to} & \quad Ax \leq b
\end{align*}
\]

With *strictly convex* assumption, we have

**Dual QP**

\[
\begin{align*}
\max & \quad -\frac{1}{2}(p' + \alpha'A)Q^{-1}(A'\alpha + p) - \alpha'b \\
\text{subject to} & \quad \alpha \geq 0
\end{align*}
\]