Constrained Convex Optimization

Master DSC/MLDM/CPS2

Ievgen Redko

ievgen.redko@univ-st-etienne.fr

Laboratoire Hubert Curien, UMR CNRS 5516
Université Jean Monnet Saint-Étienne
amaury.habrard@univ-st-etienne.fr

Semester 2

Outline

Some Course material

- The textbook for the optimization part is *Convex Optimization* (Boyd & Vandenberghe).
- It is freely available in PDF on Boyd's website: `http://www.stanford.edu/~boyd/cvxbook/`
- This book goes a lot further than what we will cover, so you can refer to it if you want to know more about optimization.
- Online lectures from Boyd (Standford university) are freely available on the web (`http://www.stanford.edu/class/ee364a/videos.html` or youtube: `http://www.youtube.com/watch?v=McLq1hEq3UY`)
- Useful book for matrix/vector operations, derivatives, ... : the matrix cookbook `http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/3274/pdf/imm3274.pdf`

Plan

What is it?

(Mathematical) Optimization

Find values for variables such that a given function is minimized (or maximized), sometimes under constraints. Standard form:

$$\begin{aligned} \min_{x} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad 1 \leq i \leq m \\ & h_i(x) = 0, \quad 1 \leq i \leq p \end{aligned}$$

What is it?

(Mathematical) Optimization

Find values for variables such that a given function is minimized (or maximized), sometimes under constraints. Standard form:

$$\begin{aligned} \min_{x} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad 1 \leq i \leq m \\ & h_i(x) = 0, \quad 1 \leq i \leq p \end{aligned}$$

- f_0 is the **objective function**, $x = (x_1, \ldots, x_n)$ the **variables** and $f_1, \ldots, f_m, h_1, \ldots, h_p$ defines $m + p$ **inequality and equality constraints** on the variables.
- Optimal solution x^* has smallest value of f_0 among all vectors that satisfy the constraints.
- An optimization problem can be **constrained** or **unconstrained**.

What for?

- **In our lab**. Optimizing the setting of a femto-second laser to improve its performance.

What for?

- **In our lab**. Optimizing the setting of a femto-second laser to improve its performance.
- **Economics**. Optimization of agent behavior (expected profit). Game theory. Portfolio optimization.

What for?

- **In our lab**. Optimizing the setting of a femto-second laser to improve its performance.
- **Economics**. Optimization of agent behavior (expected profit). Game theory. Portfolio optimization.
- **Control engineering**. Online optimization of robot behavior.

What for?

- **In our lab**. Optimizing the setting of a femto-second laser to improve its performance.
- **Economics**. Optimization of agent behavior (expected profit). Game theory. Portfolio optimization.
- **Control engineering**. Online optimization of robot behavior.
- **Machine learning**. Data fitting. Parameter inference, e.g., in Support Vector Machines.

What for?

- **In our lab**. Optimizing the setting of a femto-second laser to improve its performance.
- **Economics**. Optimization of agent behavior (expected profit). Game theory. Portfolio optimization.
- **Control engineering**. Online optimization of robot behavior.
- **Machine learning**. Data fitting. Parameter inference, e.g., in Support Vector Machines.
- **Computer Vision**. Image segmentation and restoration. Dictionary learning. 2D/3D shape matching/recovery. Tracking.

Continuous functions

Definition (Continuous function)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is **continuous** at some point $c \in \mathbb{R}^n$ if

$$\lim_{x \to c} f(x) = f(c).$$

Intuitively, it means that "small" changes in the input x result in "small" changes in the output $f(x)$.

Definition (Differentiable function, Gradient)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is **differentiable** if its derivative exists at all $x \in \mathbb{R}^n$. Then the **gradient** of f at x is the vector whose components are the partial derivatives of f at x:

$$\nabla f(x) = \left(\frac{\partial_f}{\partial_{x_1}}, \cdots, \frac{\partial_f}{\partial_{x_n}} \right)^T$$

Differentiable functions

Definition (Differentiable function, Gradient)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is **differentiable** if its derivative exists at all $x \in \mathbb{R}^n$. Then the **gradient** of f at x is the vector whose components are the partial derivatives of f at x:

$$\nabla f(x) = \left(\frac{\partial_f}{\partial_{x_1}}, \cdots, \frac{\partial_f}{\partial_{x_n}} \right)^T$$

Theorem

If f is differentiable then it is continuous. The converse is false (for instance, $f(x) = |x|$).

Twice differentiable and smooth functions

Definition (Twice differentiable function, Hessian)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is **twice differentiable** if its second-order derivative exists at all $x \in \mathbb{R}^n$. Then the **Hessian matrix** of f at x is the matrix whose components are the partial second-order derivatives of f at x:

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2_f}{\partial^2_{x_1}} & \frac{\partial^2_f}{\partial_{x_1}\partial_{x_2}} & \cdots & \frac{\partial^2_f}{\partial_{x_1}\partial_{x_n}} \\ \frac{\partial^2_f}{\partial_{x_2}\partial_{x_1}} & \frac{\partial^2_f}{\partial^2_{x_2}} & \cdots & \frac{\partial^2_f}{\partial_{x_2}\partial_{x_n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2_f}{\partial_{x_n}\partial_{x_1}} & \frac{\partial^2_f}{\partial_{x_n}\partial_{x_2}} & \cdots & \frac{\partial^2_f}{\partial^2_{x_n}} \end{bmatrix}$$

Twice differentiable and smooth functions

Definition (Twice differentiable function, Hessian)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is **twice differentiable** if its second-order derivative exists at all $x \in \mathbb{R}^n$. Then the **Hessian matrix** of f at x is the matrix whose components are the partial second-order derivatives of f at x:

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2_f}{\partial^2_{x_1}} & \frac{\partial^2_f}{\partial_{x_1}\partial_{x_2}} & \cdots & \frac{\partial^2_f}{\partial_{x_1}\partial_{x_n}} \\ \frac{\partial^2_f}{\partial_{x_2}\partial_{x_1}} & \frac{\partial^2_f}{\partial^2_{x_2}} & \cdots & \frac{\partial^2_f}{\partial_{x_2}\partial_{x_n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2_f}{\partial_{x_n}\partial_{x_1}} & \frac{\partial^2_f}{\partial_{x_n}\partial_{x_2}} & \cdots & \frac{\partial^2_f}{\partial^2_{x_n}} \end{bmatrix}$$

Definition (Smooth function)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is **smooth** if it has derivatives of all orders.

Positive semi-definiteness

Definition (Positive semi-definiteness)

A matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ is **positive semi-definite** (PSD), denoted $\mathbf{M} \succeq 0$, if all its eigenvalues are positive. Alternatively, $\mathbf{M}$ is PSD if all the following matrices have a positive determinant:

- the upper left 1-by-1 corner of $\mathbf{M}$,
- the upper left 2-by-2 corner of $\mathbf{M}$,
- $\cdots$,
- $\mathbf{M}$ itself.

Other possibility: check that for any vector $\mathbf{z} \neq 0$: $\mathbf{z}^t\mathbf{M}\mathbf{z} \geq 0$.

Positive semi-definiteness

Definition (Positive semi-definiteness)

A matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ is **positive semi-definite** (PSD), denoted $\mathbf{M} \succeq 0$, if all its eigenvalues are positive. Alternatively, $\mathbf{M}$ is PSD if all the following matrices have a positive determinant:

- the upper left 1-by-1 corner of $\mathbf{M}$,
- the upper left 2-by-2 corner of $\mathbf{M}$,
- $\cdots$,
- $\mathbf{M}$ itself.

Other possibility: check that for any vector $\mathbf{z} \neq 0$: $\mathbf{z}^t \mathbf{M} \mathbf{z} \geq 0$.

Complexity of PSD check

The complexity of checking $\mathbf{M} \succeq 0$ is $O(n^3)$. It can be done by hand for very small matrices, but when $\mathbf{M}$ gets large, it becomes costly even for a computer.

Note on PSD Matrices and eigenvalues

PSD matrix $M \Rightarrow$ eigenvalues of M are positive (easy)

If $\boldsymbol{v}$ is an eigenvector of M with eigenvalue λ; we have $M\boldsymbol{v} = \lambda\boldsymbol{v}$. Then $\boldsymbol{v}^t M\boldsymbol{v} = \boldsymbol{v}^t\lambda\boldsymbol{v}$ that is positive by assumption and thus we must have $\lambda > 0$.

PSD matrix $M \Leftarrow$ eigenvalues of M are positive (sketch)

Let $M = PDP^{-1}$ the eigenvalue decomposition of M, $P =$ matrix of (right) eigenvectors ${\boldsymbol{v}_i}_{i=1}^n$ of M and D a diagonal of eigenvalues. As eigenvectors are linearly independent, $\forall\ \boldsymbol{z}$, $\boldsymbol{z} = c_1\boldsymbol{v}_1 + \dots c_n\boldsymbol{v}_n$, with the $c_i's \in \mathbb{R}$. Thus:

$$\begin{aligned}
\boldsymbol{z}^T M\boldsymbol{z} &= (c_1\boldsymbol{v}_1^t + \ldots + c_n\boldsymbol{v}_n^t)PDP^{-1}(c_1\boldsymbol{v}_1 + \ldots + c_n\boldsymbol{v}_n) \\
&= (c_1\|\boldsymbol{v}_1\|_2^2 \ \ldots \ c_n\|\boldsymbol{v}_n\|_2^2) \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix} \begin{pmatrix} c_1\|\boldsymbol{v}_1\|_2^2 \\ c_2\|\boldsymbol{v}_2\|_2^2 \\ \vdots \\ c_n\|\boldsymbol{v}_n\|_2^2 \end{pmatrix} \\
&= \lambda_1 c_1^2 + \cdots + \lambda_n c_n^2
\end{aligned}$$

which is clearly positive since the eigenvalues of the matrix are positive. Note from that from properties of eigenvectors $\|\boldsymbol{v}_i\|_2^2 = \boldsymbol{v}_i^t \cdot \boldsymbol{v}_i = 1$ and $\boldsymbol{v}_i^t \cdot \boldsymbol{v}_j = 0$ for i$\neq j$.

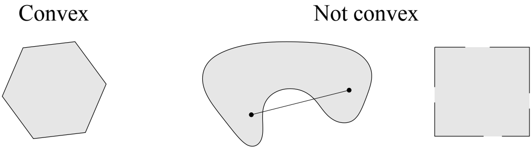
Capital letters denote matrices and small letters vectors

- $\frac{\delta \mathbf{x}^T \mathbf{a}}{\delta \mathbf{x}} = \frac{\delta \mathbf{a}^T \mathbf{x}}{\delta \mathbf{x}} = \mathbf{a}$
- $\frac{\delta \mathbf{a} \mathbf{X} \mathbf{b}}{\delta \mathbf{X}} = \mathbf{a}\mathbf{b}^T$
- $\frac{\delta \mathbf{a} \mathbf{X}^T \mathbf{b}}{\delta \mathbf{X}} = \mathbf{b}\mathbf{a}^T$
- $\frac{\delta}{\delta \mathbf{X}} Tr(\mathbf{X}) = \mathbf{I}$
- $\frac{\delta}{\delta \mathbf{X}} Tr(\mathbf{X}\mathbf{A}) = \mathbf{A}^T$
- $\frac{\delta}{\delta \mathbf{X}} Tr(\mathbf{A}\mathbf{X}\mathbf{B}) = \mathbf{A}^T\mathbf{B}^T$
- $\frac{\delta}{\delta \mathbf{X}} Tr(\mathbf{X}^T\mathbf{A}) = \mathbf{A}$
- $\ldots \rightarrow$ matrix cookbook (https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf)

Definition (Convex set)

A **convex set** C contains line segment between any two points in the set.

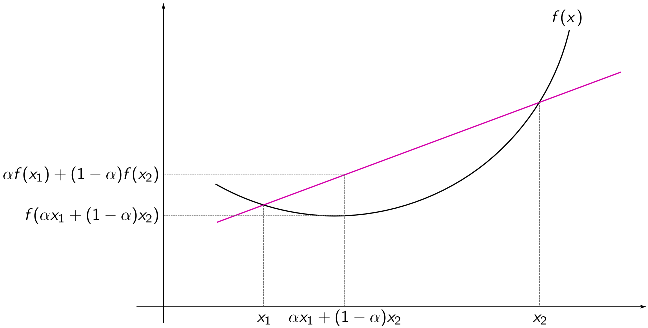
$$x_1, x_2 \in C, \quad 0 \le \alpha \le 1 \quad \Longrightarrow \quad \alpha x_1 + (1-\alpha)x_2 \in C$$



Convex functions

Definition (Convex function)

$f : \mathbb{R}^n \to \mathbb{R}$ is a **convex function** if $x_1, x_2 \in \mathbb{R}^n, 0 \le \alpha \le 1$
$\implies f(\alpha x_1 + (1-\alpha)x_2) \le \alpha f(x_1) + (1-\alpha) f(x_2)$



Can also see it as "the area above the curve is a convex set".

- f **strictly convex** if strict inequality (unique stationary point).
- f is (strictly) **concave** if $-f$ is (strictly) convex.

Convex functions ctd

- f **strictly convex** if strict inequality (unique stationary point).
- f is (strictly) **concave** if $-f$ is (strictly) convex.
- Examples of convex functions
  - linear: $ax + b$ on $\mathbb{R}$ for any $a, b \in \mathbb{R}$
  - exponential: $\exp^{ax}$ for any $a \in \mathbb{R}$
  - powers: x^a on $\mathbb{R}_{++}$, for $a \geq 1$ or $a \leq 0$
  - powers of absolute value: $|x|^p$ for $p \geq 1$
  - negative entropy: $x \log x$ on $\mathbb{R}_+$
  - norms: $\|x\|_p$ for $p \geq 1$; $\|x\|_\infty = \max_k |x_k|$

- f **strictly convex** if strict inequality (unique stationary point).
- f is (strictly) **concave** if $-f$ is (strictly) convex.
- Examples of convex functions
  - linear: $ax + b$ on $\mathbb{R}$ for any $a, b \in \mathbb{R}$
  - exponential: $\exp^{ax}$ for any $a \in \mathbb{R}$
  - powers: x^a on $\mathbb{R}_{++}$, for $a \geq 1$ or $a \leq 0$
  - powers of absolute value: $|x|^p$ for $p \geq 1$
  - negative entropy: $x \log x$ on $\mathbb{R}_+$
  - norms: $\|x\|_p$ for $p \geq 1$; $\|x\|_\infty = \max_k |x_k|$
- Examples of concave functions
  - linear
  - powers: x^a on $\mathbb{R}_{++}$, for $0 \leq a \leq 1$
  - logarithm: $\log x$ on $\mathbb{R}_{++}$

Establishing the convexity of a function

How to establish the convexity of f?

Establishing the convexity of a function

How to establish the convexity of f?

1. **Plot** to see what it looks like

How to establish the convexity of f?

1. **Plot** to see what it looks like
2. Verify **definition** (easier if f restricted to a single line)

Establishing the convexity of a function

How to establish the convexity of f?

1. **Plot** to see what it looks like
2. Verify **definition** (easier if f restricted to a single line)
3. Use the **PSD property** of the Hessian: f twice differentiable is convex iff $\nabla^2 f(x) \succeq 0 \quad \forall x$.

Establishing the convexity of a function

How to establish the convexity of f?

1. **Plot** to see what it looks like
2. Verify **definition** (easier if f restricted to a single line)
3. Use the **PSD property** of the Hessian: f twice differentiable is convex iff $\nabla^2 f(x) \succeq 0 \quad \forall x$.
4. Express f with other convex functions using operations that preserve convexity:
    - nonnegative weighted sum of convex functions,
    - composition of a convex function with a linear function,
    - pointwise maximum and supremum of convex functions,
    - composition of convex functions,
    - minimization of convex function over a convex set.

Local and global optima

Definition (Local optimum)

We say that x is a **local minimum** (resp. maximum) of f if there exists $R > 0$ such that $f(x) \leq f(z)$ (resp. $f(x) \geq f(z)$) for all z satisfying $\|z - x\|_2 \leq R$. Moreover we have $\nabla^2 f(x) \succeq 0$. (resp. $\nabla^2 f(x) \preceq 0$).

Definition (Local optimum)

We say that x is a **local minimum** (resp. maximum) of f if there exists $R > 0$ such that $f(x) \leq f(z)$ (resp. $f(x) \geq f(z)$) for all z satisfying $\|z - x\|_2 \leq R$. Moreover we have $\nabla^2 f(x) \succeq 0$. (resp. $\nabla^2 f(x) \preceq 0$).

Intuition

It means that x achieves the best value within a neighboring set of solutions.

Local and global optima

Definition (Local optimum)

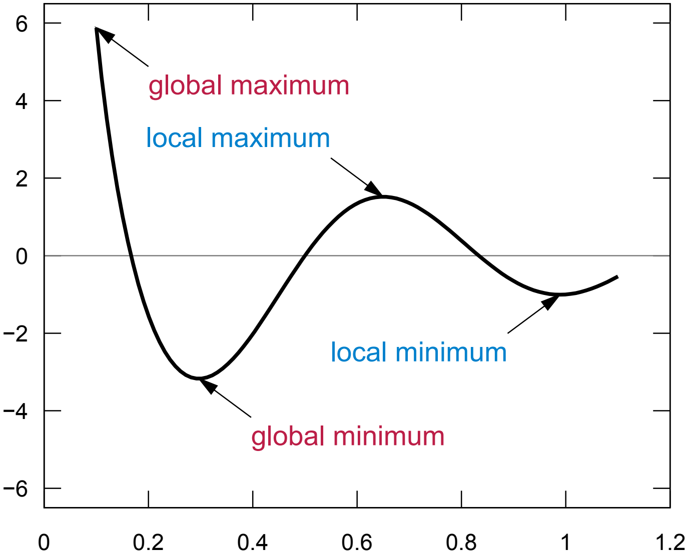
We say that x is a **local minimum** (resp. maximum) of f if there exists $R > 0$ such that $f(x) \leq f(z)$ (resp. $f(x) \geq f(z)$) for all z satisfying $\|z - x\|_2 \leq R$. Moreover we have $\nabla^2 f(x) \succeq 0$. (resp. $\nabla^2 f(x) \preceq 0$).

Intuition

It means that x achieves the best value within a neighboring set of solutions.

Definition (Global optimum)

We say that x^* is a **global minimum** (resp. maximum) of f if $f(x^*) \leq f(z)$ (resp. $f(x^*) \geq f(z)$) for all $z \in \mathbb{R}^n$.


6
4
2
0
−2
−4
−6
0
0.2
0.4
0.6
0.8
1
1.2
global maximum
local maximum
local minimum
global minimum

Key properties of convex optimization

Theorem (Key property!!!)

If f is convex (resp. concave), then any local minimum (resp. maximum) is a global minimum (resp. maximum).

Key properties of convex optimization

Theorem (Key property!!!)

If f is convex (resp. concave), then any local minimum (resp. maximum) is a global minimum (resp. maximum).

Theorem (Uniqueness of global optimum)

If f is **strictly** *convex (resp. concave), then there is a unique global minimum (resp. maximum).*

Key properties of convex optimization

Theorem (Key property!!!)

If f is convex (resp. concave), then any local minimum (resp. maximum) is a global minimum (resp. maximum).

Theorem (Uniqueness of global optimum)

If f is **strictly** *convex (resp. concave), then there is a unique global minimum (resp. maximum).*

Convention

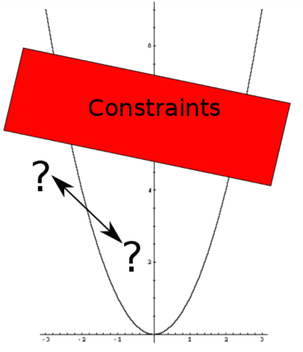
Without loss of generality, we consider minimization problems (since maximizing a concave function f is equivalent to minimizing the convex function $-f$).

Solving Convex Optimization Problems

Gradient descent-based algorithms

With a well-tuned algorithms we are able to find (or accurately approximate) the global optimum

How to deal with constraints?



Plan

Constrained optimization

- We are now interested in smooth **constrained** convex optimization, i.e., in **standard form**:

$$\begin{aligned} \min_{x} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad 1 \leq i \leq m \\ & h_i(x) = 0, \quad 1 \leq i \leq p) \end{aligned}$$

where $f_0, f_1, \ldots, f_m$ are smooth convex functions (and $h_1, \ldots, h_p$ are linear functions).

Constrained optimization

- We are now interested in smooth **constrained** convex optimization, i.e., in **standard form**:

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad 1 \leq i \leq m \\ & h_i(x) = 0, \quad 1 \leq i \leq p) \end{aligned}$$

  where $f_0, f_1, \dots, f_m$ are smooth convex functions (and $h_1, \dots, h_p$ are linear functions).
- Many special cases:
  - **Linear Programming** (LP): $\forall i,\ f_i$ linear.
    Specific algorithm: **Simplex**.
  - **Quadratic Programming** (QP): f_0 quadratic, $\{f_i\}_{i=1}^m$ linear.
  - **Quadratically Constrained Quadratic Programming** (QCQP): $f_0, f_1, \dots, f_m, h_1, \dots, h_p$ quadratic.
  - ...

Constrained optimization is **challenging**!

- The problem may not be **feasible**, i.e., the **feasibility set**

$$F = \{x \mid f_i(x) \leq 0,\ 1 \leq i \leq m\} = \emptyset,$$

  meaning that no point satisfies the constraints.
- The minimum of f_0 may violate the constraints.
- Therefore, adapting Gradient Descent for constrained optimization **is not trivial**!

General solution

A general-purpose solution: **interior point algorithms**.

- Solve **general smooth convex constrained** problems.
- Solve **large problems** (many variables, many constraints).
- **Reliable** and **efficient**.

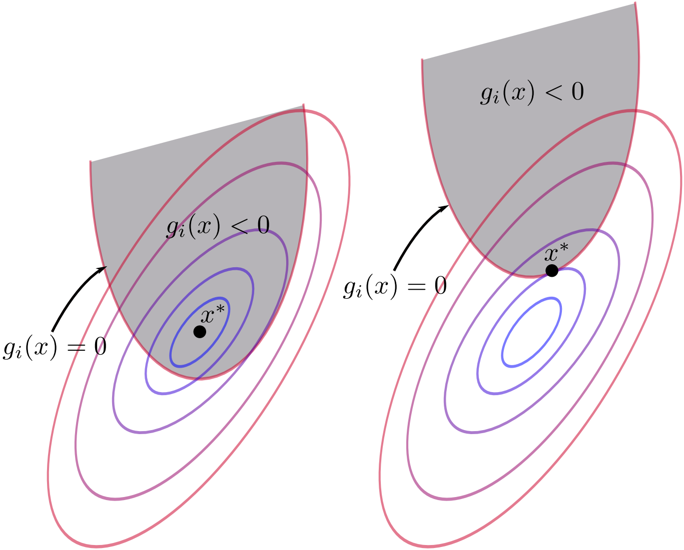
General solution

A general-purpose solution: **interior point algorithms**.

- Solve **general smooth convex constrained** problems.
- Solve **large problems** (many variables, many constraints).
- **Reliable** and **efficient**.

As many other methods, based on **duality theory**.

Graphical illustration



Lagrangian and **Duality Theory**

Basic idea: *take the constraints into account by* **augmenting the objective function** *with a weighted sum of the constraint functions.*

Constrained optimization

Standard form (not necessarily convex)

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad 1 \leq i \leq m \\ & h_i(x) = 0, \quad 1 \leq i \leq p \end{aligned}$$

variable $x \in \mathbb{R}^n$, domain D, optimal value p^*

Lagrangian: $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)$$

- weighted sum of objective and constraint functions
- λ_i is the Lagrange multiplier associated with $f_i(x) \leq 0$
- ν_i is the Lagrange multiplier associated with $h_i(x) = 0$.

Definition (Lagrange dual function $g : \mathbb{R}^m \to \mathbb{R}$)

We now define the **Lagrange dual function** $g : \mathbb{R}^m \to \mathbb{R}$ as the minimum value of the Lagrangian over x, i.e., for $\lambda \in \mathbb{R}^m$ and $\nu \in \mathbb{R}^p$

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in D} L(x, \lambda, \nu) \\ &= \inf_{x \in D} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{m} \nu_i h_i(x) \right). \end{aligned}$$

g is concave, can be $-\infty$ for some λ, ν

Property of the Lagrange Dual Function

Theorem (Property)

The dual function yields a lower bound on the optimal objective value p^* of the original problem, i.e., for any $\lambda \succeq 0$, we have

$$g(\lambda, \nu) \leq p^*.$$

Proof:

if $\hat{x}$ is feasible, for any $\lambda \succeq 0$:

$f_0(\hat{x}) \geq L(\hat{x}, \lambda, \nu) \geq \inf_{x \in D} L(x, \lambda, \nu) = g(\lambda, \nu)$

minimizing over all feasible $\hat{x}$ gives $p^* \geq g(\lambda, \nu)$.

Exercise 1: Least-norm solution of Linear Equation

$$\begin{aligned} \min_{x} \quad & x^T x = \|x\|_2^2 \\ \text{subject to} \quad & Ax = b \end{aligned}$$

with $x \in \mathbb{R}^d$, $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$. Note that the equality constraint corresponds actually to m constraints!

Questions

- Give the Lagrangian formulation
- Compute the gradient over x
- Compute an optimum with respect to x
- Deduce a lower bound for the solution of the optimization problem above.

Exercise 1: Solution

1. **Lagrangian**:

$$L(x,\nu) = x^T x + \nu^T (Ax - b),\ \nu \in \mathbb{R}^m$$

Exercise 1: Solution

1. **Lagrangian**:

$$L(x,\nu) = x^T x + \nu^T (Ax - b),\ \ \nu \in \mathbb{R}^m$$

2. **Gradient**:

$$\nabla_x L(x,\nu) = 2x + A^T \nu$$

Exercise 1: Solution

1. **Lagrangian**:

$$L(x, \nu) = x^T x + \nu^T (Ax - b),\ \ \nu \in \mathbb{R}^m$$

2. **Gradient**:

$$\nabla_x L(x, \nu) = 2x + A^T \nu$$

3. To minimize over x **set the gradient to zero**:

$$\nabla_x L(x, \nu) = 2x + A^T \nu = 0 \Rightarrow x = -(1/2) A^T \nu$$

Exercise 1: Solution

1. **Lagrangian**:

$$L(x, \nu) = x^T x + \nu^T (Ax - b), \ \nu \in \mathbb{R}^m$$

2. **Gradient**:

$$\nabla_x L(x, \nu) = 2x + A^T \nu$$

3. To minimize over x **set the gradient to zero**:

$$\nabla_x L(x, \nu) = 2x + A^T \nu = 0 \Rightarrow x = -(1/2) A^T \nu$$

4. Plug in L to **obtain** g:

$$g(\nu) = L((-1/2) A^T \nu, \nu) = -\frac{1}{4} \nu^T A A^T \nu - b^T \nu$$

   a concave function of ν

Exercise 1: Solution

1. **Lagrangian**:

$$L(x,\nu) = x^T x + \nu^T (Ax - b),\ \nu \in \mathbb{R}^m$$

2. **Gradient**:

$$\nabla_x L(x,\nu) = 2x + A^T \nu$$

3. To minimize over x **set the gradient to zero**:

$$\nabla_x L(x,\nu) = 2x + A^T \nu = 0 \Rightarrow x = -(1/2)A^T \nu$$

4. Plug in L to **obtain** g:

$$g(\nu) = L((-1/2)A^T \nu, \nu) = -\frac{1}{4}\nu^T A A^T \nu - b^T \nu$$

   a concave function of ν

5. **Lower bound property:**

$$p^* \geq -(1/4)\nu^T A A^T \nu - b^T \nu,\ \forall \nu$$

Exercise 2: Standard form LP

$$\begin{aligned} \min_x \quad & c^T x \\ \text{subject to} \quad & Ax = b, x \succeq 0 \\ \text{with } c \in \mathbb{R}^d, \quad & x \in \mathbb{R}^d, A \in \mathbb{R}^{m \times d}, b \in \mathbb{R}^m. \end{aligned}$$

Questions

- Give the Lagrangian formulation
- Define the dual function g
- Deduce a lower bound on the optimum p^*

Exercise 2: Solution

1. Lagrangian:

$$L(x, \lambda, \nu) = c^T x + \nu^T (Ax - b) - \lambda^T x = -b^T \nu + (c + A^T \nu - \lambda)^T x$$

Exercise 2: Solution

1. Lagrangian:

$$L(x, \lambda, \nu) = c^T x + \nu^T (Ax - b) - \lambda^T x = -b^T \nu + (c + A^T \nu - \lambda)^T x$$

2. L is affine in x thus

$$g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) = \begin{cases} -b^T \nu & \text{if } A^T \nu - \lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

   g is linear on affine domain $\{(\lambda, \nu) | A^T \nu - \lambda + c = 0\}$, hence concave.

Exercise 2: Solution

1. Lagrangian:

$$L(x,\lambda,\nu) = c^T x + \nu^T(Ax-b) - \lambda^T x = -b^T\nu + (c + A^T\nu - \lambda)^T x$$

2. L is affine in x thus

$$g(\lambda,\nu) = \inf_x L(x,\lambda,\nu) = \begin{cases} -b^T\nu & \text{if } A^T\nu - \lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

   g is linear on affine domain $\{(\lambda,\nu) | A^T\nu - \lambda + c = 0\}$, hence concave.

3. **Lower bound property:**

$$p^* \geq -b^T\nu \text{ if } A^T\nu + c \succeq 0$$

Exercise 3: Two-way partitioning

1. We want to **partition a set** of 3 tennis players x_1, x_2 and x_3 in **two clusters** given a **compatibility score**
2. We add one point when they win a double game and -1 when they loose it. Here is a matrix summarizing these scores:

$$W = \begin{pmatrix} 0 & -3 & 10 \\ -3 & 0 & 2 \\ 10 & 2 & 0 \end{pmatrix}$$

3. To solve this problem, we create a vector of 3 variables $\boldsymbol{x}^t = (x_1, x_2, x_3)^t$, s.t. $\forall i,\ x_i = \{-1, 1\}$. If x_i and x_j $(i \neq j)$ have **the same value** then they belong to the **same cluster**.

Exercise 3: Two-way partitioning problem

This task can be solved by the following optimization problem

$$\begin{aligned} \min_x \quad & x^T W x \\ \text{subject to} \quad & x_i^2 = 1, i = 1, \ldots, n \end{aligned}$$

- A **non convex** problem: feasible set contains 2^n points
- **Interpretation**: partition $\{1, \ldots, n\}$ in two sets, W_{ij} is the cost of assigning i, j to the same set, $-W_{ij}$ is cost of assigning to different sets

Goal

1. Write the dual function and interpret it
2. Deduce a general lower bound
3. Solve it with example of tennismen

1. Compute the **dual function**

$$
\begin{aligned}
g(\nu) &= \inf_x (x^T W x + \sum_i \nu_i (x_i^2 - 1)) \\
&= \inf_x x^T (W + \boldsymbol{diag}(\nu)) x - \mathbf{1}^T \nu \Rightarrow
\end{aligned}
$$

$$
g(\nu) = \begin{cases} -\mathbf{1}^T \nu & \text{if } W + \boldsymbol{diag}(\nu) \succeq 0 \\ -\infty & \text{otherwise} \end{cases}
$$

Exercise 3 - Two-way partitioning - solution

1. Compute the **dual function**

$$g(\nu) = \inf_x (x^T W x + \sum_i \nu_i (x_i^2 - 1))$$

$$= \inf_x x^T (W + \boldsymbol{diag}(\nu)) x - \mathbf{1}^T \nu \Rightarrow$$

$$g(\nu) = \begin{cases} -\mathbf{1}^T \nu & \text{if } W + \boldsymbol{diag}(\nu) \succeq 0 \\ -\infty & \text{otherwise} \end{cases}$$

2. **Lower bound property:**

$$p^* \geq -\mathbf{1}^T \nu \text{ if } W + \boldsymbol{diag}(\nu) \succeq 0$$

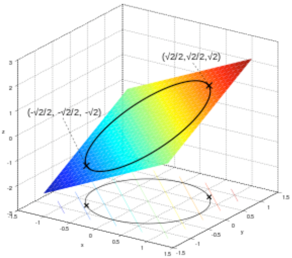
Example: $\nu = \max(W)\mathbf{1}$ gives $p^* \geq -n \max(W)$ if $W + \boldsymbol{diag}(\nu) \succeq 0$.

3. **Rewrite** the optimization problem

$$\min_{\boldsymbol{x}} \boldsymbol{x}^T W \boldsymbol{x} \quad = \quad \min_{\boldsymbol{x}} -6x_1x_2 + 20x_1x_3 + 4x_2x_3$$

The constraints $x_i^2 = 1$ imposes each $x_i = \{-1, 1\}$. An optimum of -30 is given by $x_1 = 1$, $x_2 = 1$ and $x_3 = -1$ (or equivalently $x_1 = -1$, $x_2 = -1$ and $x_3 = 1$) meaning that we group x_1 and x_2 together and we let x_3 alone.

$$\min_{x,y} f(x,y) = x + y \text{ s.t. } x^2 + y^2 = 1$$



Goal

1. Write the Lagrangian function
2. Compute the gradient of the Lagrangian
3. Deduce the solution of the problem.

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

2. Set $\nabla L(x, y, \lambda) = 0$:

Exercise 4 - Lagrange multipliers - solution

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

2. Set $\nabla L(x, y, \lambda) = 0$:

$$\frac{\partial L}{\partial x} = 0 \Leftrightarrow 1 + 2\lambda x = 0$$

Exercise 4 - Lagrange multipliers - solution

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

2. Set $\nabla L(x, y, \lambda) = 0$:

$$\frac{\partial L}{\partial x} = 0 \Leftrightarrow 1 + 2\lambda x = 0$$
$$\frac{\partial L}{\partial y} = 0 \Leftrightarrow 1 + 2\lambda y = 0$$

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

2. Set $\nabla L(x, y, \lambda) = 0$:

$$\frac{\partial L}{\partial x} = 0 \Leftrightarrow 1 + 2\lambda x = 0$$
$$\frac{\partial L}{\partial y} = 0 \Leftrightarrow 1 + 2\lambda y = 0$$
$$\frac{\partial L}{\partial \lambda} = 0 \Leftrightarrow x^2 + y^2 - 1 = 0$$

Exercise 4 - Lagrange multipliers - solution

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

2. Set $\nabla L(x, y, \lambda) = 0$:

$$\frac{\partial L}{\partial x} = 0 \Leftrightarrow 1 + 2\lambda x = 0$$
$$\frac{\partial L}{\partial y} = 0 \Leftrightarrow 1 + 2\lambda y = 0$$
$$\frac{\partial L}{\partial \lambda} = 0 \Leftrightarrow x^2 + y^2 - 1 = 0$$

3. This yields

$$x == \frac{1}{2\lambda} \text{ and } y == \frac{1}{2\lambda}$$

Exercise 4 - Lagrange multipliers - solution

1. Write the Lagrangian

$$L(x, y, \lambda) = x + y + \lambda(x^2 + y^2 - 1)$$

2. Set $\nabla L(x, y, \lambda) = 0$:

$$\frac{\partial L}{\partial x} = 0 \Leftrightarrow 1 + 2\lambda x = 0$$
$$\frac{\partial L}{\partial y} = 0 \Leftrightarrow 1 + 2\lambda y = 0$$
$$\frac{\partial L}{\partial \lambda} = 0 \Leftrightarrow x^2 + y^2 - 1 = 0$$

3. This yields

$$x == \frac{1}{2\lambda} \text{ and } y == \frac{1}{2\lambda}$$

4. Substituting into the last equation yields $\frac{1}{4\lambda^2} + \frac{1}{4\lambda^2} = 1$, so $\lambda = 1/\sqrt{2}$, which implies that the optimum is defined by

$$x^* = -\sqrt{2}/2 \text{ and } y^* = -\sqrt{2}/2$$

We know that **dual function** *gives a* **lower bound** *on the* **optimal objective value***. Natural question: what is the* **best lower bound***?*

We know that **dual function** *gives a* **lower bound** *on the* **optimal objective value**. *Natural question: what is the* **best lower bound**?

Definition (Lagrange dual problem)

This leads to the **Lagrange dual problem**:

$$\begin{aligned} \max_{\lambda,\nu} \quad & g(\lambda,\nu) \\ \text{subject to} \quad & \lambda \succeq 0 \end{aligned}$$

We know that **dual function** *gives a* **lower bound** *on the* **optimal objective value**. *Natural question: what is the* **best lower bound**?

Definition (Lagrange dual problem)

This leads to the **Lagrange dual problem**:

$$\begin{aligned} \max_{\lambda,\nu} \quad & g(\lambda,\nu) \\ \text{subject to} \quad & \lambda \succeq 0 \end{aligned}$$

- The dual problem is convex (g is concave + convex constraints). The optimal value is denoted by d^*
- λ, ν is dual feasible if $\lambda \succeq 0$, $(\lambda,\nu) \in \boldsymbol{dom}(g)$
- Original problem is called **primal problem**.

Theorem (Weak duality)

Simple but important inequality:

$$d^* \leq p^*.$$

Duality theory: weak duality

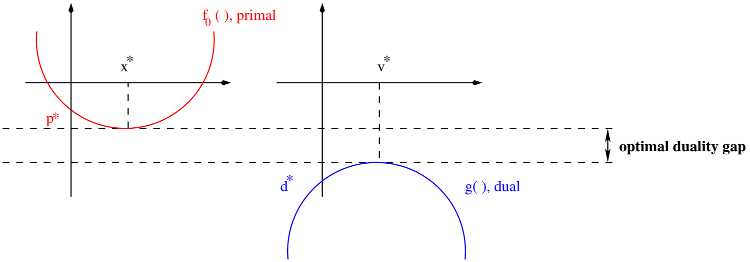
Theorem (Weak duality)

Simple but important inequality:

$$d^* \le p^*.$$

- **Always** holds (for convex and non convex problem)
- We define the **optimal duality gap** as $p^* - d^*$.
- Can be used to find a **lower bound** on the optimal value of a **non convex** problem.

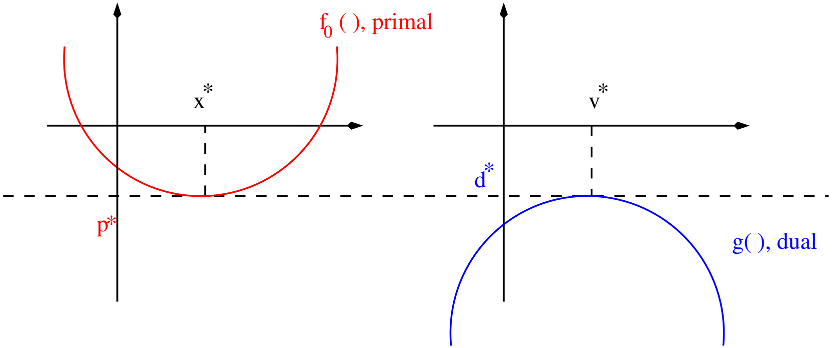
Weak duality - illustration



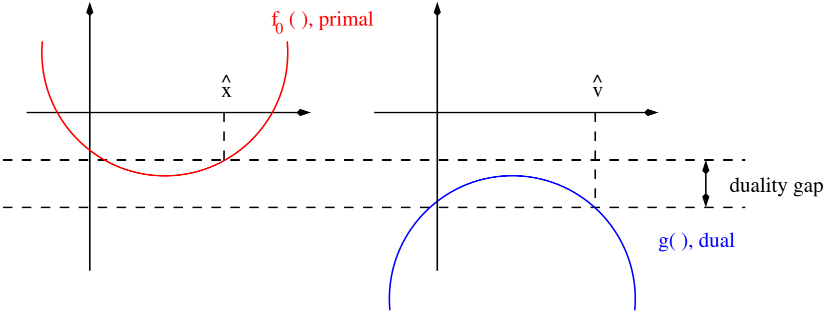
- We have $g(\nu^*, \lambda^*) = d^* \leq p^* = f(x^*)$
- Any value of g is a lower bound on p^* and the best lower bound is d^*.

- In our case, we are interested in **strong duality**: $d^* = p^*$ (optimal duality gap is zero).
- Strong duality does **not** hold in general.
- But if primal problem is convex, it usually does!
- Many results establish conditions (beyond convexity) under which strong duality holds.

Strong duality - illustration



- We have $d^* = p^*$ (optimal duality gap is zero).



- The duality gap ensures that
  $0 \leq f(\hat{x}) - f(x^*) \leq |f(\hat{\nu}) - f(\hat{x})|$

Slater's condition

Strong duality holds for a **convex problem**

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, i = 1, \ldots, n \\ & Ax = b \end{aligned}$$

If it is **strictly feasible**: $\exists x : f_i(x) < 0, i = 1, \ldots, n, \qquad Ax = b.$

- Guarantees that the dual optimum is attained (if $p^* > -\infty$)

Example: Inequality form LP

$$\begin{array}{ll} \min\limits_x & c^T x \\ \text{subject to} & Ax \succeq b \end{array}$$

1. **Dual function**

$$g(\lambda) = \inf_x((c + A^T\lambda)^T x - b^T\lambda) = \begin{cases} -b^T\lambda & \text{if } A^T\lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

2. **Dual problem**

$$\begin{array}{ll} \max\limits_\lambda & -b^T\lambda \\ \text{subject to} & A^T\lambda + c = 0, \lambda \succeq 0 \end{array}$$

3. From **Slater's condition**: $p^* = d^*$ if $A\hat{x} \succeq b$ for some $\hat{x}$
4. In fact here $p^* = d^*$ except when primal and dual are infeasible

Complementary slackness

If **strong duality holds**, then for primal optimal x^* and dual optimal (λ^*, ν^*) we obtain:

$$
\begin{aligned}
f_0(x^*) = g(\lambda^*, \nu^*) \quad &= \quad \inf_x \left(f_0(x) + \sum_{i=1}^{m} \lambda_i^* f_i(x) + \sum_{i=1}^{p} \nu_i^* h_i(x) \right) \\
&\leq \quad f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* f_i(x^*) + \sum_{i=1}^{p} \nu_i^* h_i(x^*) \\
&\leq \quad f_0(x^*)
\end{aligned}
$$

Hence, the last two lines **hold** with **equality**!

- x^* minimizes $L(x, \lambda^*, \nu^*)$
- Complementary slackness:

$$\lambda_i > 0 \Rightarrow f_i(x^*) = 0 \text{ and } f_i(x^*) < 0 \Rightarrow \lambda_i^* = 0$$

Given **differentiable** f_0, f_i, h_i, *the following four conditions are called* **KKT conditions**

1. *Primal constraints:* $f_i(x) \leq 0$, $i = 1, \ldots, m$ *and* $h_i(x) = 0, i = 1, \ldots, p$
2. *Dual constraints* $\lambda \succeq 0$
3. **Complementary slackness***:* $\lambda_i f_i(x) = 0, i = 1 \ldots, m$
4. **Gradient** *of Lagrangian with respect to* x **vanishes***:*

$$\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \sum_{i=1}^{p} \nu_i \nabla h_i(x) = 0$$

If **strong duality** *holds and* x, λ, ν *are* **optimal***, then they must satisfy the* **KKT conditions***.*

KKT conditions for convex problems

Convex problems

If $\hat{x}, \hat{\lambda}, \hat{\nu}$ *satisfy KKT for a convex problem, then:*

1. *They are* **optimal**
2. *Complementary slackness implies*

$$f_0(\hat{x}) = L(\hat{x}, \hat{\lambda}, \hat{\nu})$$

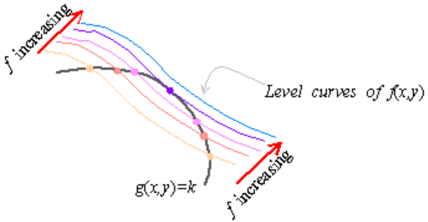
3. *Vanishing gradient and convexity imply:*

$$g(\hat{\lambda}, \hat{\nu}) = L(\hat{x}, \hat{\lambda}, \hat{\nu})$$

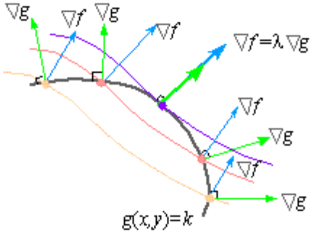
4. $f_0(\hat{x}) = g(\hat{\lambda}, \hat{\nu}) \Rightarrow$ **sufficient** *optimality conditions.*

An illustration with an equality constraint

1. Draw level curves of $f(x, y)$ and equality constraints $g(x, y) = k$.



2. **Optimal solution** is where their **gradients are colinear**.



from `http://math.etsu.edu/multicalc/prealpha/chap2/chap2-9/part1.htm`

Example: Water-filling problem

Assume $\alpha_i > 0$ and consider the following problem:

$$\begin{array}{ll} \min\limits_{x} & -\sum_{i=1}^{n} \log(x_i + \alpha_i) \\ \text{subject to} & x \succeq 0, \mathbf{1}^T x = 1 \end{array}$$

Task

1. Write the Lagrangian
2. Deduce optimal values with respect to KKT conditions

Example: solution

1. Langrangian writes

$$L(x,\lambda,\nu) = -\sum_{i=1}^{n} \log(x_i + \alpha_i) - \sum_{i=1} \lambda_i x_i + \nu(\mathbf{1}^T x - 1)$$

2. x^* is optimal iff $x^* \succeq 0$, $\mathbf{1}^T x^* = 1$ and there exist $\lambda \in \mathbb{R}^n$, $\nu \in \mathbb{R}$ such that:

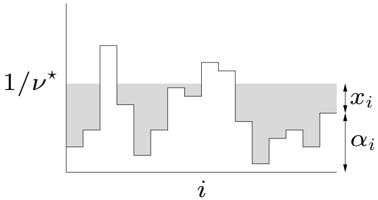
$$\lambda \succeq 0, \lambda_i x_i^* = 0, \frac{1}{x_i^* + \alpha_i} + \lambda_i = \nu$$

3. Complementary slackness can be rewritten as

$$x_i^*(\nu - 1/(\alpha_i + x_i^*)) = 0 \Rightarrow \frac{1}{x_i^* + \alpha_i} + \lambda_i = \nu \Rightarrow \frac{1}{x_i^* + \alpha_i} \leq \nu$$

4. From KKT:
   - If $\nu < 1/\alpha_i$ then $x_i^* > 0$ and last condition implies
     $$x_i^* = 1/\nu - \alpha_i$$
   - If $\nu \geq 1/\alpha_i$ then $x_i^* = 0$ otherwise complementary slackness is violated. Thus
     $$x_i^* = \max\{0, 1/\nu - \alpha_i\}$$
   - We determine ν from
     $$\mathbf{1}^T x^* = \sum_{i=1}^{n} \max\{0, 1/\nu - \alpha_i\} = 1$$

Interpretation of the result

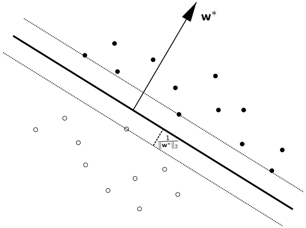


- n patches; level of patch i is at height α_i
- Flood area with unit amount of water
- Resulting level is $1/\nu^*$

SVM hard-margin primal form

Given a training set $T = \{\mathbf{z_i} = (\mathbf{x_i}, \ell_i)\}_{i=1}^n$ of linearly-separable instances ($\mathbf{x_i} \in \mathbb{R}^d, \ell_i \in \{-1, 1\}$). We want to maximize the margin $\frac{1}{\|\mathbf{w}^*\|_2}$. The largest-margin hyperplane $(\mathbf{w}^*, b^*)$ separating the instances of T is the solution of the following optimization problem.

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|_2^2$$
$$\text{s.t. } \forall i, \quad \ell_i(\langle \mathbf{w}, \mathbf{x_i}\rangle + b) \geq 1$$



Towards the dual

- The Lagrangian of the problem is:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{n} \alpha_i [1 - \ell_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b)],$$

where $\boldsymbol{\alpha} \succeq 0$.

- The Lagrangian of the problem is:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{n} \alpha_i[1 - \ell_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b)],$$

  where $\boldsymbol{\alpha} \succeq 0$.

- Lagrange dual function $g(\boldsymbol{\alpha})$ is obtained by minimizing L over $\mathbf{w}$ and b. To do that, we set the derivatives of L w.r.t. $\mathbf{w}$ and b to 0:

$$\begin{cases} \frac{\partial L(\mathbf{w},b,\boldsymbol{\alpha})}{\partial \mathbf{w}} & = & \mathbf{w} - \sum_{i=1}^{n} \ell_i \alpha_i \mathbf{x_i} = 0 \\ \frac{\partial L(\mathbf{w},b,\boldsymbol{\alpha})}{\partial \mathbf{b}} & = & \sum_{i=1}^{n} \ell_i \alpha_i = 0 \end{cases}$$

- We thus have:

$$\mathbf{w} = \sum_{i=1}^{n} \ell_i \alpha_i \mathbf{x_i} \quad \text{and} \quad \sum_{i=1}^{n} \ell_i \alpha_i = 0.$$

- We thus have:

$$\mathbf{w} = \sum_{i=1}^{n} \ell_i \alpha_i \mathbf{x_i} \quad \text{and} \quad \sum_{i=1}^{n} \ell_i \alpha_i = 0.$$

- Substitute in L to get the dual function $g(\boldsymbol{\alpha})$ to maximize:

$$\begin{aligned}
L(\mathbf{w}, b, \boldsymbol{\alpha}) &= \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{n} \alpha_i [1 - \ell_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b)] \\
&= \frac{1}{2} \sum_{i,j=1}^{n} \ell_i \ell_j \alpha_i \alpha_j \langle \mathbf{x_i}, \mathbf{x_j} \rangle \\
&\quad - \sum_{i,j=1}^{n} \ell_i \ell_j \alpha_i \alpha_j \langle \mathbf{x_i}, \mathbf{x_j} \rangle - b \sum_{i=1}^{n} \ell_i \alpha_i + \sum_{i=1}^{n} \alpha_i \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \ell_i \ell_j \alpha_i \alpha_j \langle \mathbf{x_i}, \mathbf{x_j} \rangle \\
&= g(\boldsymbol{\alpha}).
\end{aligned}$$

SVM hard-margin dual form

$$\max_{\boldsymbol{\alpha}} \qquad g(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \ell_i \ell_j \alpha_i \alpha_j \langle \mathbf{x_i}, \mathbf{x_j} \rangle$$

$$\text{s.t.} \qquad \sum_{i=1}^{n} \ell_i \alpha_i = 0, \;\; \boldsymbol{\alpha} \succeq 0$$

SVM hard-margin dual

SVM hard-margin dual form

$$\max_{\boldsymbol{\alpha}} \qquad g(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \ell_i \ell_j \alpha_i \alpha_j \langle \mathbf{x_i}, \mathbf{x_j} \rangle$$

$$\text{s.t.} \qquad \sum_{i=1}^{n} \ell_i \alpha_i = 0, \ \boldsymbol{\alpha} \succeq 0$$

- This problem is **convex** (duality theory).
- **Optimal** value of the **dual and the primal coincide**.
- The largest margin can be recovered by taking $\mathbf{w}^* = \sum_{i=1}^{n} \ell_i \alpha_i^* \mathbf{x_i}$.

What about b^*?

Understanding the optimal solution

1. From KKT conditions, we get:

$$\alpha_i^*[\ell_i(\langle \mathbf{w}^*, \mathbf{x_i} \rangle + b^*) - 1] = 0, \quad 1 \leq i \leq n.$$

- **Case 1**: $\ell_i(\langle \mathbf{w}^*, \mathbf{x_i} \rangle + b^*) > 1$.
  $\mathbf{x_i}$ is not on the margin and $\alpha_i^* = 0$.
- **Case 2**: $\ell_i(\langle \mathbf{w}^*, \mathbf{x_i} \rangle + b^*) = 1$.
  $\mathbf{x_i}$ is on the margin and $\alpha_i^* \neq 0$. From these points we can deduce b^*.

Understanding the optimal solution

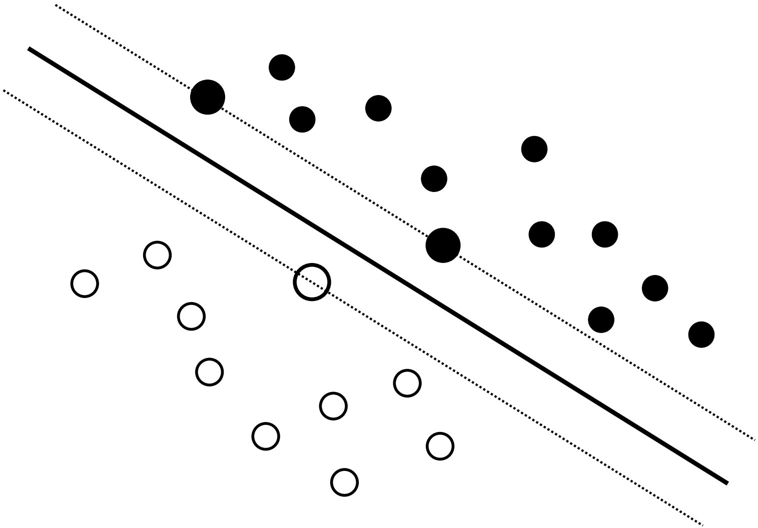
1. From KKT conditions, we get:

$$\alpha_i^*[\ell_i(\langle \mathbf{w}^*, \mathbf{x_i} \rangle + b^*) - 1] = 0, \quad 1 \leq i \leq n.$$

   - **Case 1**: $\ell_i(\langle \mathbf{w}^*, \mathbf{x_i} \rangle + b^*) > 1$.
     $\mathbf{x_i}$ is not on the margin and $\alpha_i^* = 0$.
   - **Case 2**: $\ell_i(\langle \mathbf{w}^*, \mathbf{x_i} \rangle + b^*) = 1$.
     $\mathbf{x_i}$ is on the margin and $\alpha_i^* \neq 0$. From these points we can deduce b^*.

2. Therefore, $\mathbf{w}^*$ is defined **only** in terms of the training instances that **lie on the margin**. We call these points **support vectors**.

Support vectors: illustration



Plan

Context

Optimization problem

$$\begin{array}{ll} \min\limits_{x} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, i = 1, \ldots m \end{array}$$

Assumptions

- f_0 convex and f_i are **convex**, **twice continuously differentiable**
- Problem is **strictly feasible**: $\exists \hat{x} : f_i(\hat{x}) < 0, i = 1, \ldots m$
- **Strong duality** holds and dual optimum is attained.
- **No equality constraints** (equivalent to two inequality constraints $h_i(x) < 0$ and $-h_i(x) < 0$ (or $h_i(x) - \epsilon < 0$ and $-h_i(x) - \epsilon < 0$ for a small ϵ to remain feasible)

- **Main idea** is to solve the constrained problem by **solving a sequence of unconstrained problems**.
- Duality provides an **exact stopping criteria**.
- We consider a simple-to-implement **barrier method**.
- As for now, we assume that we are given a **strictly feasible starting point** x (talk later on how to find it).

Interior point methods: log barrier

Goal

Approximately formulate *the constrained problem as an unconstrained problem*

Solution

Consider the following approximation:

$$\min_{x} \quad f_0(x) + \sum_{i=1}^{m} -(1/t)\log(-f_i(x)).$$

- **Convex** and **differentiable** problem.
- The function $\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x))$ is called the **log barrier**.
- Its domain is **the set of points** that **strictly satisfy the constraints** of the original problem.
- It **grows without bound** if $f_i(x) \to 0$ for any i ("barrier"), no matter the value of the positive parameter t.

Interior point methods: log barrier ctd

Solution

$$\min_x \quad f_0(x) + \sum_{i=1}^{m} -(1/t)\log(-f_i(x)).$$

- The approximation gets better as t gets larger (intuitive).
- If t is too large, function and derivatives vary very quickly near the boundary of the feasible set causing numerical problems.

Solution

$$\min_{x} \quad f_0(x) + \sum_{i=1}^{m} -(1/t)\log(-f_i(x)).$$

- The approximation gets better as t gets larger (intuitive).
- If t is too large, function and derivatives vary very quickly near the boundary of the feasible set causing numerical problems.

Implementation

- Increase parameter t at each step and start each minimization at the solution of previous problem.
- For $t > 0$, the **central path** is defined as the set of solutions $x^*(t)$, $t > 0$, which we call the **central points**.

- Each central point has the following properties:
  1. It is **strictly feasible**.
  2. It **yields** a **dual feasible point**, and hence a **lower bound** on the optimal objective value p^*, with duality gap m/t.
- Property 2 gives the **stopping criterion** by telling us how far we are from optimal:

$$f_0(x^*(t)) - p^* \leq m/t,$$

Interior point methods: algorithm

Barrier algorithm: given a strictly feasible x, $t = t^0 > 0$, $\mu > 1$, tolerance $\epsilon > 0$.

1. Compute $x^*(t)$ by minimizing (67), starting at x.
2. Update: $x = x^*(t)$.
3. If $m/t < \epsilon$, stop.
4. Otherwise, iterate with $t = \mu t$.

Wait! How do I find a **strictly feasible** x **to start** with?

Interior point methods: find a strictly feasible point

Consider the following problem:

$$\begin{aligned} \min_{x,s} \quad & s \\ \text{subject to} \quad & f_i(x) \leq s, \quad 1 \leq i \leq m \end{aligned}$$

- s can be seen as a bound on the "maximum infeasibility of the problem". The goal is to drive it below zero.
- This problem is **always strictly feasible**!
  Indeed, can pick any x and $s \geq \max_{i=1,\ldots,m} f_i(x)$. So what?

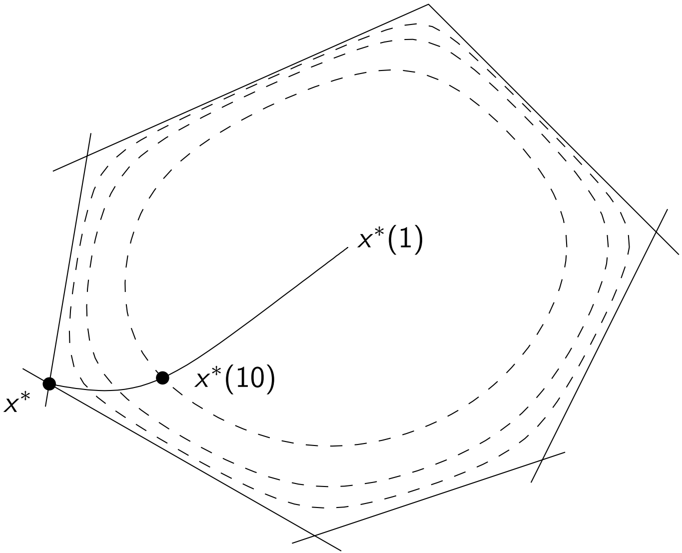
We can apply the barrier method!

Interior point methods: final algorithm

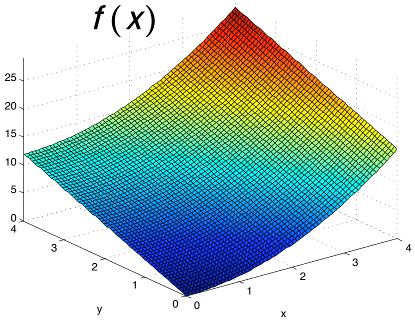
Barrier algorithm: given $t = t^0 > 0$, $\mu > 1$, tolerance $\epsilon > 0$.

- Phase I: find a strictly feasible starting point x (or declare infeasibility) by solving the maximum infeasibility problem.
- Phase II:
  1. Compute $x^*(t)$ by minimizing (67), starting at x.
  2. Update: $x = x^*(t)$.
  3. If $m/t < \epsilon$, stop.
  4. Otherwise, iterate with $t = \mu t$.

Interior point methods: illustration

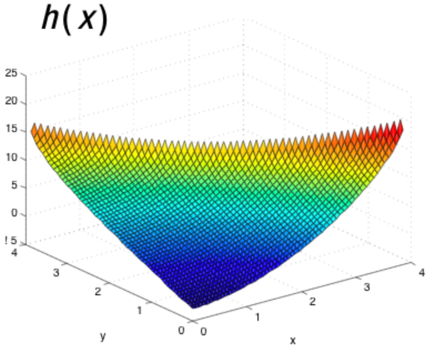


$$\min_{x \in \mathbb{R}^{\not\models}} \quad f_0(x) = x^2 + 3 * y$$
$$\text{subject to} \quad -x - y + 4 \geq 0$$



Example - interior point formulation

$$\min_{x \in \mathbb{R}^2} \quad h(x) = x^2 + 3 * y - \log(-x - y + 4)$$



A barrier is created along the boundary of the inequality constraint $-x - y + 4 = 0$.

Plan

Summary

- Solving constrained problems builds on tools such as **duality theory**.

Summary

- Solving constrained problems builds on tools such as **duality theory**.
- Another approach is to transform it to an **unconstrained problem**.

Summary

- Solving constrained problems builds on tools such as **duality theory**.
- Another approach is to transform it to an **unconstrained problem**.
- Interior point algorithms (such as the **barrier method**).

Summary

- Solving constrained problems builds on tools such as **duality theory**.
- Another approach is to transform it to an **unconstrained problem**.
- Interior point algorithms (such as the **barrier method**).

Take-home message #1

Convexity and **smoothness** help!

Summary

- Solving constrained problems builds on tools such as **duality theory**.
- Another approach is to transform it to an **unconstrained problem**.
- Interior point algorithms (such as the **barrier method**).

Take-home message #1

Convexity and **smoothness** help!

Take-home message #2

Duality and **optimality** conditions are **essential**!

Some examples/exercises

Example 1: Duality

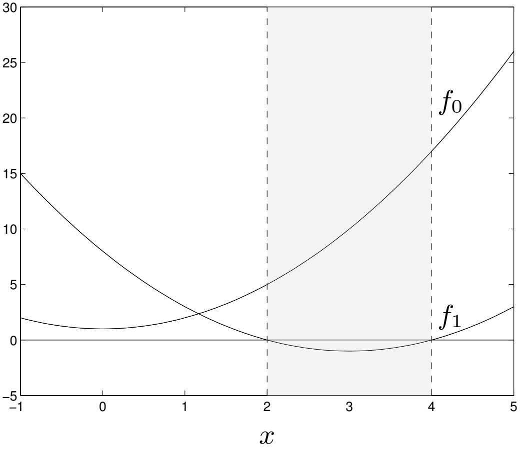
Consider the simple optimization problem

$$\begin{array}{rc} \min\limits_{x} & f_0(x) = x^2 + 1 \\ \text{subject to} & f_1(x) = (x-2)(x-4) \leq 0, \end{array}$$

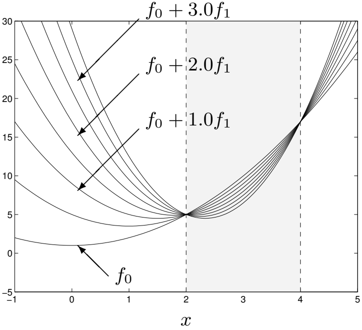
Questions

1. Analyse primal problem: **feasible set, optimal value and optimal solutions**.
2. Lagrangian and dual functions. Plot f_0 and **feasible set**, plot the **Lagrangian** with different λ's (*e.g.* 1, 2, 3). Write the dual function.
3. State the dual problem. Find the dual optimal solution. Does **strong duality** hold?

The feasible set is interval $[2, 4]$. The unique optimal point is $x^* = 2$, optimal value is $p^* = 5$. The plot shows f_0 and f_1.



The Lagrangian:
$L(x, \lambda) = x^2 + 1 + \lambda(x-2)(x-4) = (1+\lambda)x^2 - 6\lambda x + 1 + 8\lambda$



- For all λ, $p^* \geq \inf_x L(x, \lambda) = g(\lambda)$
- $g(\lambda)$ increases when $\lambda \in [0; 2)$, and then decreases as $\lambda > 2$.
- We have $p^* = g(\lambda)$ for $\lambda = 2$

Dual function:

$$g(\lambda) = \inf_x (1+\lambda)x^2 - 6\lambda x + 1 + 8\lambda$$

- Taking the derivative of $g(x,\lambda)$ with respect to x we have for $\lambda > -1$:

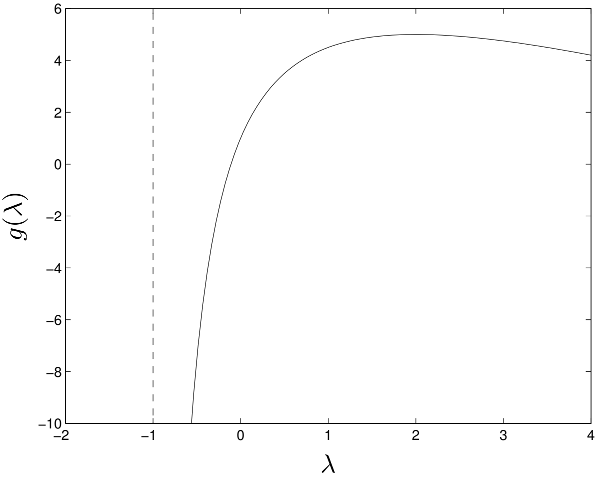
$$\begin{aligned} \frac{\partial L(x,\lambda)}{\partial x} = 0 \quad &\Leftrightarrow \quad 2(1+\lambda)x - 6\lambda = 0 \\ &\Leftrightarrow \quad x = \frac{3\lambda}{1+\lambda} \end{aligned}$$

- For $\lambda < -1$ the dual function is unbounded below. Then the dual is given by:

$$g(\lambda) = \begin{cases} -9\lambda^2/(1+\lambda) + 1 + 8\lambda & \text{if } \lambda > -1 \\ -\infty & \text{if } \lambda \leq -1 \end{cases}$$

Example 1: Solution

$$g(\lambda) = \begin{cases} -9\lambda^2/(1+\lambda) + 1 + 8\lambda & \text{if } \lambda > -1 \\ -\infty & \text{if } \lambda \le -1 \end{cases}$$



The Lagrange **dual problem** is:

$$\begin{aligned} \max_{\lambda} \quad & -9\lambda^2/(1+\lambda)+1+8\lambda \\ \text{subject to} \quad & \lambda \geq 0 \end{aligned}$$

- $\frac{\partial g(\lambda)}{\partial \lambda} = 0$ is equivalent to $\lambda^2 + 2\lambda - 8 = 0$ that admits 2 solutions $\lambda_1 = -4$ and $\lambda_2 = 2$.
- The constraint tells us that the optimum we are looking for is attained for $\lambda = 2$ and $d^* = g(2) = 5$.

Strong duality holds!

Example 2: Optimization

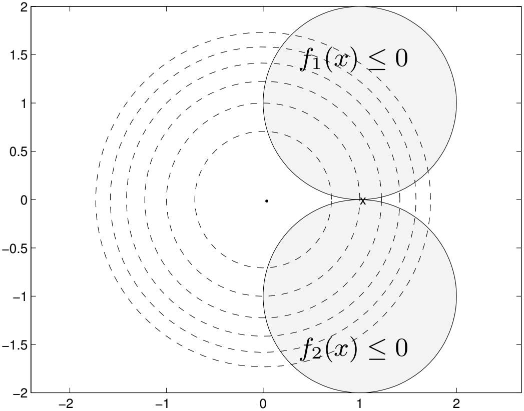
Consider the following $QCQP$

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^2} \quad & x_1^2 + x_2^2 \\ \text{subject to} \quad & (x_1 - 1)^2 + (x_2 - 1)^2 \leq 1 \\ & (x_1 - 1)^2 + (x_2 + 1)^2 \leq 1 \end{aligned}$$

Questions

1. Try to **guess** the optimal point x^* and its optimal value p^*
2. Give the **KKT conditions**. Do there exist Lagrange multipliers that give optimal x^*?
3. Derive and solve the Lagrange **dual problem**. Does **strong duality** hold?

Example 2: Solution



- The figure shows the **feasible sets** (the 2 shaded disks) and some contour lines of the **objective function**.
- There is **only one feasible point** $(1, 0)$, so it is **optimal** for the primal problem and we have $p^* = 1$.

Lagrangian

$$x_1^2 + x_2^2 + \lambda_1((x_1-1)^2 + (x_2-1)^2 - 1) + \lambda_2((x_1-1)^2 + (x_2+1)^2 - 1)$$

KKT conditions

- **Primal constraints:**

$$(x_1-1)^2 + (x_2-1)^2 \leq 1, \quad (x_1-1)^2 + (x_2+1)^2 \leq 1$$

- **Dual constraints:** $\lambda_1 \geq 0$, $\lambda_2 \geq 0$
- **Complementary slackness:**

$$\lambda_1((x_1-1)^2 + (x_2-1)^2 - 1) = \lambda_2((x_1-1)^2 + (x_2+1)^2 - 1) = 0$$

- **Gradient of the Lagrangian is 0:**

$$\begin{aligned} 2x_1 + 2\lambda_1(x_1-1) + 2\lambda_2(x_1-1) &= 0 \\ 2x_2 + 2\lambda_1(x_2-1) + 2\lambda_2(x_2+1) &= 0 \end{aligned}$$

At $\boldsymbol{x} = (1,0)$ these conditions reduce to $\lambda_1 \geq 0$, $\lambda_2 \geq 0$, $2 = 0$, $-2\lambda_1 + 2\lambda_2 = 0$, which clearly have **no solution**.

Example 2: Lagrange Dual

- **Dual function**

$$g(\lambda_1, \lambda_2) = \inf_{x_1, x_2} L(x_1, x_2, \lambda_1, \lambda_2)$$

where

$$\begin{aligned}
&L(x_1, x_2, \lambda_1, \lambda_2) \\
&\quad = \quad x_1^2 + x_2^2 + \lambda_1((x_1 - 1)^2 + (x_2 - 1)^2) + \lambda_2((x_1 - 1)^2 + (x_2 + 1)^2 - 1) \\
&\quad = \quad (1 + \lambda_1 + \lambda_2)(x_1^2 + x_2^2) - 2(\lambda_1 + \lambda_2)x_1 - 2(\lambda_1 - \lambda_2)x_2 + \lambda_1 + \lambda_2
\end{aligned}$$

- **Gradient** of L w.r.t. x_1 and x_2 is equal to 0 for

$$x_1 = \frac{\lambda_1 + \lambda_2}{1 + \lambda_1 + \lambda_2}$$
$$x_2 = \frac{\lambda_1 - \lambda_2}{1 + \lambda_1 + \lambda_2}$$

implying

$$g(\lambda_1, \lambda_2) = \begin{cases} -\frac{(\lambda_1+\lambda_2)^2+(\lambda_1-\lambda_2)^2}{1+\lambda_1+\lambda_2} + \lambda_1 + \lambda_2 & \text{if } 1 + \lambda_1 + \lambda_2 \geq 0 \\ -\infty & \text{otherwise} \end{cases}$$

Example 2: Lagrange Dual

The Lagrange **dual problem** is given by:

$$\begin{array}{ll} \max\limits_{\boldsymbol{\lambda}\in\mathbb{R}^2} & (\lambda_1+\lambda_2-(\lambda_1-\lambda_2)^2)/(1+\lambda_1+\lambda_2) \\ \text{subject to} & \lambda_1, \lambda_2 \geq 0. \end{array}$$

- Since g is **symmetric**, optimum occurs when $\lambda_1 = \lambda_2$ (if it exists, check the derivatives). The dual function then simplifies to

$$g(\lambda_1, \lambda_2) = \frac{2\lambda_1}{2\lambda_1 + 1}$$

- $g(\lambda_1, \lambda_2)$ tends to 1 as $\lambda_1 \to \infty$. We have thus $d^* = p^* = 1$ but the dual is **not attained**.
- KKT conditions only if (1) strong duality holds, (2) the primal optimum is attained and (3) the dual optimum is attained. In this example the KKT conditions **fail** because the **dual** optimum is **not attained**.

Example 3 - Optimization and KKT

We consider the following problem

$$\begin{array}{ll} \min_{\boldsymbol{x}\in\mathbb{R}^3} & -3x_1^2 + x_2^2 + 2x_3^2 + 2(x_1 + x_2 + x_3) \\ \text{subject to} & x_1^2 + x_2^2 + x_3^2 = 1. \end{array}$$

Questions

1. Derive the **KKT conditions**.
2. Find all **solutions** x and ν that satisfy these conditions.
3. What is the **optimum**?

The **Lagrangian** is:

$$L(\boldsymbol{x}, \nu) = -3x_1^2 + x_2^2 + 2x_3^2 + 2(x_1 + x_2 + x_3) + \nu(x_1^2 + x_2^2 + x_3^2 - 1)$$

- We have

$$\begin{aligned} \frac{\partial L}{\partial x_1} &= 2((-3+\nu)x_1 + 1), \\ \frac{\partial L}{\partial x_2} &= 2((1+\nu)x_2 + 1), \\ \frac{\partial L}{\partial x_3} &= 2((2+\nu)x_3 + 1) \end{aligned}$$

- The KKT conditions are:
  1. **Constraint** of the **primal** $x_1^2 + x_2^2 + x_3^2 = 1$
  2. **Vanishing gradient**

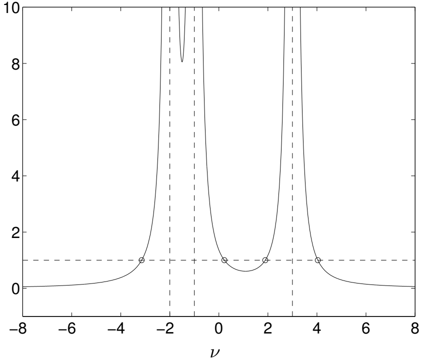
$$(-3+\nu)x_1 + 1 = 0, (1+\nu)x_2 + 1 = 0, (2+\nu)x_3 + 1 = 0$$

Example 3: Solution

- $\nu \neq -2$, $\nu \neq -1$ and $\nu \neq 3$.
- We can then express x_1, x_2 and x_3 and plug them in the constrained leading to:

$$\frac{1}{(-3+\nu)^2} + \frac{1}{(1+\nu)^2} + \frac{1}{(2+\nu)^2} = 1$$

- Analyze the lefthand side plotted below



Example 3: Solution

There are four (approximate) solutions:

$$\nu = -3.15, \nu = 0.22, \nu = 1.89, \nu = 4.04$$

Corresponding to:

$$\begin{aligned} \boldsymbol{x} &= (0.16, 0.47, -0.87), & \boldsymbol{x} &= (0.36, -0.82, -0.45) \\ \boldsymbol{x} &= (0.90, -0.35, -0.26), & \boldsymbol{x} &= (-0.97, -0.20, -0.17) \end{aligned}$$

Example 3: Solution

- Compare the values of the **objective function**:

$$f_0(\boldsymbol{x}) = 1.17, f_0(\boldsymbol{x}) = 0.67, f_0(\boldsymbol{x}) = -0.56, f_0(\boldsymbol{x}) = -4.70$$

- The **minimizer** is $\nu^* = 4.04$
- x^* is a minimizer of $L(\boldsymbol{x}, \nu^*)$ so we must have:

$$\nabla^2 f_0(x^*) + \nu^* \nabla^2 f_1(x^*) \succeq 0$$

$$\begin{bmatrix} -3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} + \nu^* \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \succeq 0$$

and therefore $\nu^* \geq 3$. So the **optimum** is given by $\nu^* = 4.04$.

Note that this problem is not convex, but strong duality holds

We consider the following LP problem

$$\begin{array}{ll} \min\limits_{\boldsymbol{x}} & x_2 \\ \text{subject to} & x_1 \leq x_2, 0 \leq x_2. \end{array}$$

What happens to this problem is we apply the barrier method?

Nothing good!

- We have the following modified unconstrained problem

$$\min_{\boldsymbol{x}} \quad x_2 - \frac{1}{t}(\log(x_2 - x_1) + \log(x_2))$$

This function is unbounded below (letting $x_1 \to -\infty$)

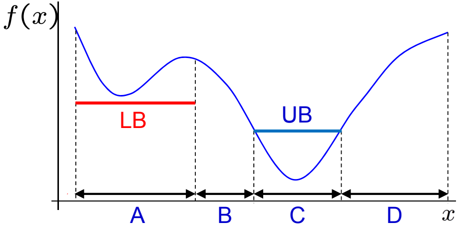
This problem is not properly defined!

Plan

Non-convex optimization

- **Hard** setting in general: **no guarantee** for the optimum.
- Unfortunately, **occurs** quite **often** (deep learning).
- Sometimes **behaves nicely**
  every local minima can be global minima
  or local minima are very close to the global one
- General **solutions exist**
  1. Grid search: uniform grid space covering
  2. Branch and bound
  3. Multiple coverings
  4. Stochastic optimization: simulated annealing, stochastic gradient (machine learning)

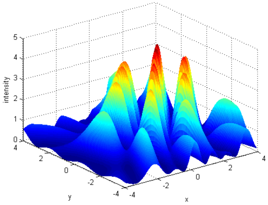
Branch and bound illustration

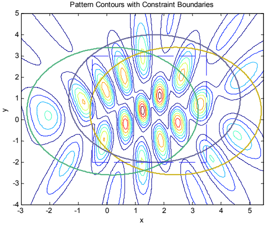


Key idea

- **Split** region into **sub-regions** and **compute bounds**
- Consider two regions A and C
- If lower bound of A is greater than upper bound of C then A can be discarded
- **Divide** (branch) regions and **repeat**

Multiple covering

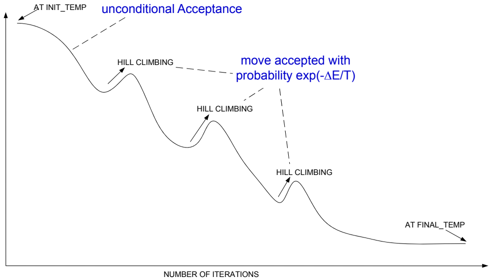




Key idea

Cover the parameter space with overlapping regions to deal with local optima, and then take advantage of efficient continuous optimization for each region

- Use multiple starting points
- Continuous optimization method for each
- Record optimum for each starting point



Key idea

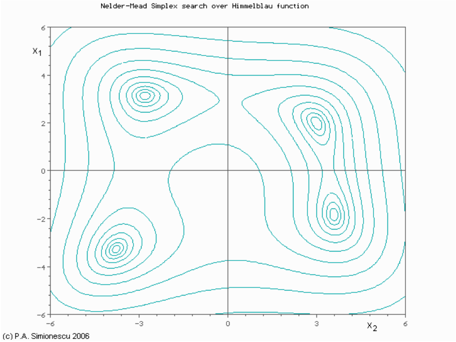
- At each iteration propose a move in the parameter space
- If the move decreases the cost, then accept it
- If the move increases the cost by ΔE, then
  - Accept it with probability $\exp\left(-\Delta E/T\right)$, or do not move.
- Adapt T to reduce the probability to move at the end.

Other Strategies

- Stochastic Gradient Descent (Machine Learning)
- Detect saddle point and try to go further along some directions
- **Approximate** the optimum thanks to a **convex surrogate**.

What if f is **not differentiable**?

- **Very annoying** as most efficient methods rely on differentiability.
- **Hard** because "we don't know in which direction to go".
- Still, **there exists** methods to deal with it (DFO: Derivative-Free Optimization).



Plan

Formulation of optimization problems

- An important **practical** problem in optimization.
- As we've seen, there are efficient algorithms to solve smooth (un)constrained convex optimization.
- So you want to **cast your problem as a smooth convex problem** (whenever possible).
- Also, some solvers do not support expressions such as $|x|$ or $\max x$. Must reformulate these.
- This is not always easy, but there are **classic tricks** out there to help you do that.
- The purpose of this section is to review some of those.

Top priorities

When formulating your problem, you should keep in mind two things.

Priority #1: smoothness and convexity

Whenever possible, prefer a **smooth and convex** formulation (even if it has a large number of variables/constraints).

Priority #2: keep the problem as small as possible

Less variables and/or less constraints means that your problem will be solved faster. Also, **sparse constraints** (i.e., including only a fraction of the problem's variables) can be handled more efficiently by solvers.

Strict inequality constraints

Using strict inequality constraints may be tempting

Consider the following problem:

$$\begin{aligned} \min_x \quad & x \\ \text{s.t.} \quad & x > 0 \end{aligned}$$

Strict inequality constraints

Using strict inequality constraints may be tempting

Consider the following problem:

$$\begin{aligned} \min_x \quad & x \\ \text{s.t.} \quad & x > 0 \end{aligned}$$

- This problem has no solution (unbounded below)!
- Illustration of why you should **never** use them: a number can get very close to zero without being exactly zero (up to numerical precision).
- Some solvers actually treat strict inequalities as nonstrict to avoid such problems.

Absolute value

Meet absolute value function $|x|$

- The absolute value is **convex** but **not differentiable** at 0.
- Some solvers **do not support** absolute value.

What to do if you have it in the objective function?

1. **Solution 1 (classic and slow)**:
   - Introduce two variables $x^+, x^- \geq 0$.
   - Express $x = x^+ - x^-$, then $|x| = x^+ + x^-$.
   - For minimization, $x^+ = 0$ (x negative) or $x^- = 0$ (x positive).
2. **Solution 2 (modern and fast)**[1]:
   - Replace $\|w\|_1 = \sum_{i=1}^{d} |w_i| = \min_{\eta \geq 0} \frac{1}{2} \sum_{i=1}^{d} \{ \frac{w_i^2}{\eta_i} + \eta_i \}$
   - Alternating minimization wrt to η and $\mathbf{w}$

Can be used to express the L_1-norm without absolute value

[1] Micchelli and Pontil, 2006; Rakotomamonjy et al. 2008)

Original problem

$$\min_{x} \quad f_0(x) + \|x\|_1$$

1. **Solution 1**

$$\begin{array}{ll} \min\limits_{x, x_+, x_-} & f_0(x) + x_+ + x_- \\ \text{subject to} & x_+ - x_- = x, \\ & x_+ \geq 0, x_- \geq 0 \end{array}$$

2. **Solution 2**

$$\begin{array}{ll} \min\limits_{x, \eta} & f_0(x) + \frac{1}{2} \sum_{i=1}^{d} \{ \frac{x_i^2}{\eta_i} + \eta_i \} \\ \text{subject to} & \eta \geq 0 \end{array}$$

Meet maximum function

- It is **convex** but **not differentiable**.
- Some solvers **do not like it** just as the absolute value function.

What to do if you have it in the objective function?

- **Reformulation**: introduce a new variable M:

Before	After
$\begin{array}{ll} \min\limits_{x} & \max_{i=1}^n x_i \\ \text{s.t.} & f_i(x) \le 0, \quad 1 \le i \le m \end{array}$	$\begin{array}{lll} \min\limits_{x,M} & M & \\ \text{s.t.} & f_i(x) \le 0, & 1 \le i \le m \\ & x_i \le M, & 1 \le i \le n \end{array}$

Can be used to express the L_∞-norm without $\max$

Simplification

- You should always try to keep the problem as **simple** as possible and avoid **redundancies** in the constraints.
- Consider the following problem:

$$\begin{aligned} \min_{x} \quad & x_1^2 + x_2^2 \\ \text{s.t.} \quad & x_1/(1+x_2^2) \le 0 \\ & (x_1+x_2)^2 = 0 \end{aligned}$$

- Is it a convex formulation?

Simplification

- You should always try to keep the problem as **simple** as possible and avoid **redundancies** in the constraints.
- Consider the following problem:

$$\begin{aligned} \min_{x} \quad & x_1^2 + x_2^2 \\ \text{s.t.} \quad & x_1/(1+x_2^2) \leq 0 \\ & (x_1 + x_2)^2 = 0 \end{aligned}$$

- Is it a convex formulation?
- Find an equivalent convex (and simpler) formulation, and put in standard form.

Relaxation

- We say that a problem P2 is a **relaxation** of a problem P1 if:
  - its objective function is the same (up to some penalty),
  - and $F_1 \subseteq F_2$ (the feasibility set of P2 is larger than that of P1).
- For instance, getting rid of a constraint is a relaxation.
- Two main uses:
  1. allow some amount of (penalized) **constraint violation**.
  2. **transform a constraint** that makes the problem hard too solve (because it is noncontinuous, non differentiable and/or nonconvex) into a smooth, convex one.

Relaxation: case 1

- Consider the following problem:

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 0, \quad 1 \leq i \leq m \end{aligned}$$

  where $f_0, f_1, \ldots, f_m$ are all smooth and convex.

- The problem can be solved efficiently, but it may be infeasible because it is **too constrained**.
- Would like to **allow the constraints to be violated**, but if this happens, it should incur a **penalty**.

Relaxation: case 1 ctd

- Solution: introduce **slack variables** $\xi_1, \ldots, \xi_m$.

$$\begin{array}{rll} \min\limits_{x,\xi_1,\ldots,\xi_m} & f_0(x) + \sum_{i=1}^m \xi_i & \\ \text{s.t.} & f_i(x) \leq \xi_i, & 1 \leq i \leq m \\ & \xi_i \geq 0, & 1 \leq i \leq m \end{array}$$

- Can use other penalty functions (e.g., squared penalty).
- Famous example: Perceptron and Support Vector Machines. Want to find the hyperplane that best separates the two classes, while allowing some violation to deal with the nonseparable case.

- Classic case: integer constraints. Example:

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{s.t.} \quad & x \in \{0, 10\} \end{aligned}$$

is very hard too solve (NP-hard).

- Classic case: integer constraints. Example:

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{s.t.} \quad & x \in \{0, 10\} \end{aligned}$$

is very hard too solve (NP-hard).

- Can be relaxed as:

$$\begin{aligned} \min_x \quad & f_0(x) \\ \text{s.t.} \quad & x \in [0, 10] \end{aligned}$$

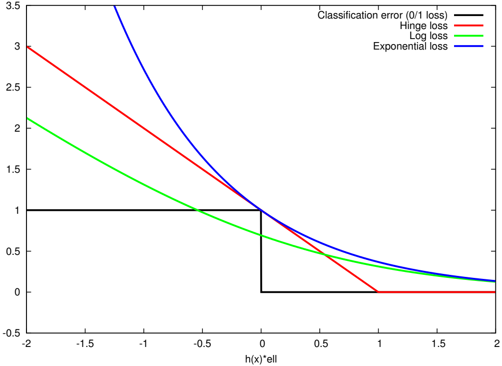
which can be solved very efficiently.

- Can then convert solution to get a (suboptimal) integer solution to the original problem (not always good!).

Surrogate function

- The function f_0 to minimize may not be smooth/convex.
- If it can't be reformulated nicely, sometimes a good **surrogate function** s for f_0 can be used:
  - s behaves similarly to the original function,
  - but has a nicer form (typically smooth convex),
- Classic examples:
  - use the L_1-norm as a convex relaxation of the L_0-norm.
  - use a convex loss function as a surrogate for the number of classification errors (0/1 loss).
- Can be combined with relaxation.

The 0/1 loss along with convex loss functions that can be used as surrogates:



Another example:
`http://www.iet.ntnu.no/~schellew/convexrelaxation/ConvexRelaxation.html`

Summary

- Many tricks can be used to reformulate problems in a nice form.
- Surprisingly many real-world problems can be expressed in a convex form one way or another.

Summary

- Many tricks can be used to reformulate problems in a nice form.
- Surprisingly many real-world problems can be expressed in a convex form one way or another.

Take-home message #1

Formulating the problem with care is an important part of optimization and must not be overlooked!

Summary

- Many tricks can be used to reformulate problems in a nice form.
- Surprisingly many real-world problems can be expressed in a convex form one way or another.

Take-home message #1

Formulating the problem with care is an important part of optimization and must not be overlooked!

Take-home message #2

Again: very small changes (e.g., L_0-norm instead of L_1-norm, integer variable instead of real variable) can turn an easy problem into a very hard one!

Plan

- When dealing with large and complex problems, the use of a well-implemented solver is recommended.
- Many solvers are available (unfortunately most of them are commercial), on many platforms.
- Here are 3 general-purpose optimization packages:
  - **MATLAB Optimization Toolbox**: many algorithms for (non)smooth (un)constrained optimization of (non)convex problems. Commercial.
  - **MOSEK**: solver that can handle many types of problems. Very efficient implementation of interior point for large-scale problems. Commercial, but provides a free unlimited size license for academic use.
  - **AMPL**: a very convenient modeling language, compatible with several solvers (including MOSEK). Commercial, but provides a student limited license (300 variables, 300 constraints). **We will use AMPL during the practical session**.

AMPL: model file

Suppose you want to solve the following problem:

$$\begin{aligned} \min_{x} \quad & 3x_1^2 + 2x_2 \\ \text{s.t.} \quad & 2x_1 + x_2 \geq 100 \\ & x_1 + x_2 \geq 80 \\ & x_1, x_2 \geq 0 \end{aligned}$$

AMPL model file

```
var x1 >= 0;
var x2 >= 0;

minimize f: 3*x1^2 + 2*x2;

subject to c1: 2*x1 + x2 >= 100;
subject to c2: x1 + x2 >= 80;
```

AMPL: features

- AMPL features:
  - it does **not** require the problem to be in standard form.
  - can minimize or maximize the objective function.
  - supports many mathematical functions in model files (e.g., sum, max, exp, etc).
  - can use multidimensional variables and sets.
  - and even more sophisticated features (ordered pairs, union of sets, etc)!
- However, some features make debugging quite complex.
- I recommend you **do not use sophisticated features unless you really have to**!

AMPL: parameters and data file

- You can also model your problem with parameters.

AMPL model file

```
var x1 >= 0;
var x2 >= 0;
param B1;
param B2;

minimize f: 3*x1^2 + 2*x2;

subject to c1: 2*x1 + x2 >= B1;
subject to c2: x1 + x2 >= B2;
```

- And use a data file to specify their value.

AMPL data file

```
param B1 := 100;
param B2 := 80;
```

AMPL: a more complex example

In this example we use a set defining the number of variables as well as the sum function, and we write all the bound constraints in a single line.

AMPL model file

```
set nbVar;
var x{i in nbVar};
param weight{i in nbVar};
param bound{i in nbVar};

minimize f: sum{i in nbVar} x[i]*weight[i];

subject to bound{i in nbVar}: x[i] >= bound[i];
```

AMPL data file

```
set nbVar := 1..5;
param weight := 1.2 2.2 0.1 3.5 2.1;
param bound := 10 40 10 20 34;
```

AMPL: homework

To save us a little bit of time during the practical section, please do the following:

- make sure you understand the examples I just showed.
- check out the following tutorial: `https://www.tu-chemnitz.de/mathematik/part_dgl/teaching/WS2009_Grundlagen_der_Optimierung/amplguide.pdf`
- A longer document `http://www.ampl.com/REFS/amplmod.pdf`
- More resources on AMPL page: *http://www.ampl.com/*, including a book that can be downloaded freely.
- We will use the free demo for the practical session `http://ampl.com/try-ampl/download-a-free-demo/` limited to 500 variables and constraints.
- You can run it freely on the NEOS server `http://ampl.com/try-ampl/run-ampl-on-neos/`

I do not expect you to install AMPL and start practicing. It's only about getting used to the syntax!

The end