direction vector

ISE 426
Optimization models and applications

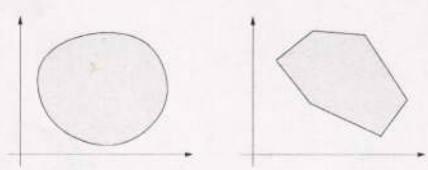
Lecture 2 — September 1, 2015

Convexity; Relaxations; Lower and upper bounds.

- Winston, chapter 1, **or**
- Winston & Venkataramanan, chapter 1, **or**
- Hillier & Lieberman, chapter 2.

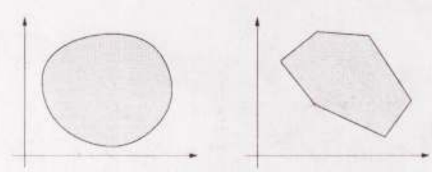
Convexity

Convex sets



Def.: A set $S \subseteq \mathbb{R}^n$ is convex if any two points x' and x'' of S are joined by a segment **entirely** contained in S:

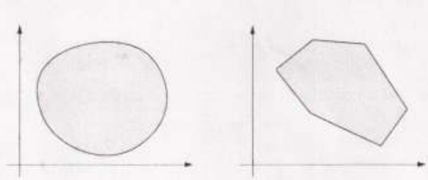
Convex sets



Def.: A set $S \subseteq \mathbb{R}^n$ is convex if any two points x' and x'' of S are joined by a segment **entirely** contained in S:

$$\forall x', x'' \in S, \alpha \in [0,1], \quad \alpha x' + (1-\alpha)x'' \in S$$

Convex sets

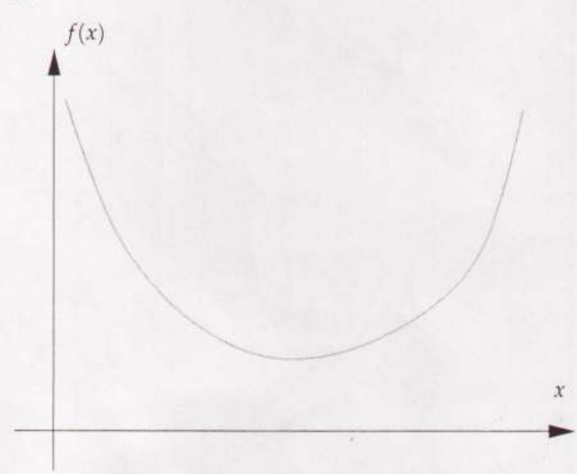


Def.: A set $S \subseteq \mathbb{R}^n$ is convex if any two points x' and x'' of S are joined by a segment **entirely** contained in S:

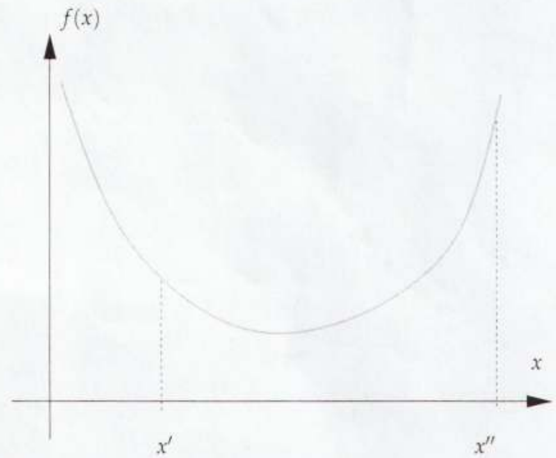
$$\forall x', x'' \in S, \alpha \in [0,1], \quad \alpha x' + (1-\alpha)x'' \in S$$

The intersection of two convex sets is convex.

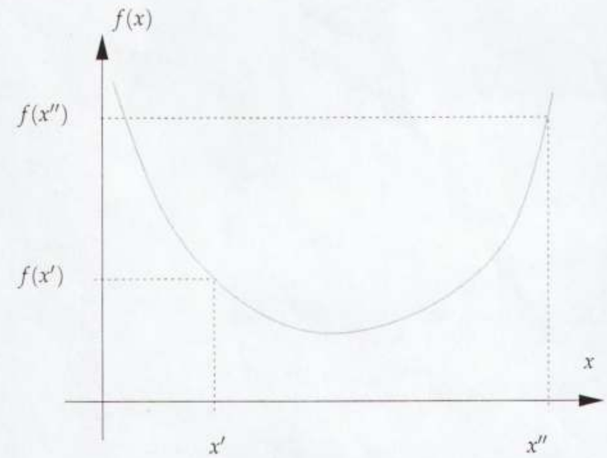
Definition



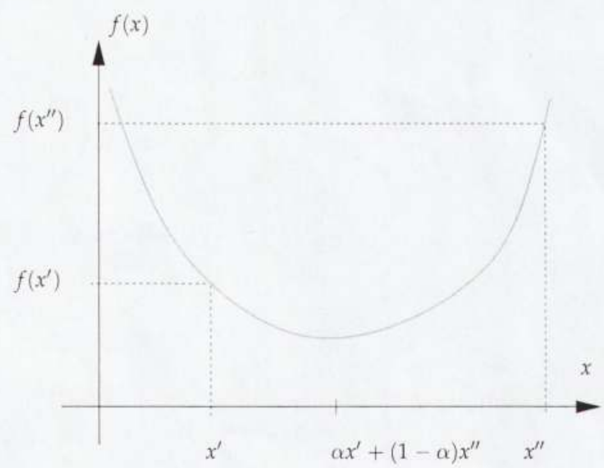
Definition



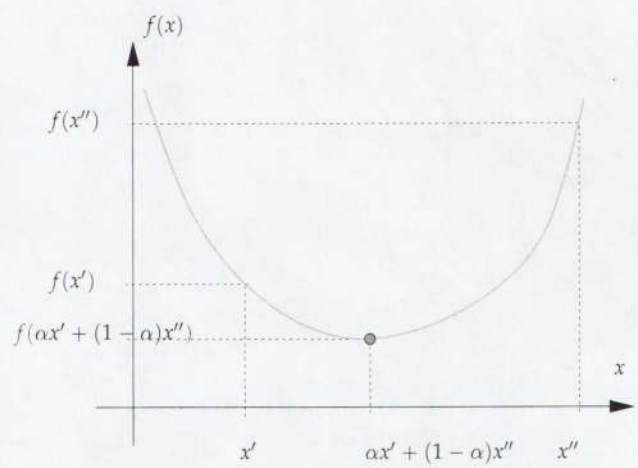
Definition



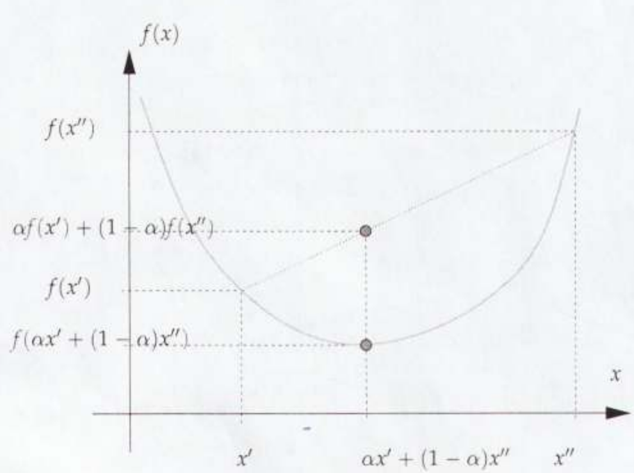
Definition



Definition



Definition



Convex functions

Def.: A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if, for any two points x' and $x'' \in \mathbb{R}^n$ and for any $\alpha \in [0,1]$

$$f(\alpha x' + (1-\alpha)x'') \quad \leq \quad \alpha f(x') + (1-\alpha)f(x'')$$

Convex functions

Def.: A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if, for any two points x' and $x'' \in \mathbb{R}^n$ and for any $\alpha \in [0, 1]$

$$f(\alpha x' + (1-\alpha)x'') \leq \alpha f(x') + (1-\alpha) f(x'')$$

- The sum of convex functions is a convex function

Convex functions

Def.: A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if, for any two points x' and $x'' \in \mathbb{R}^n$ and for any $\alpha \in [0, 1]$

$$f(\alpha x' + (1-\alpha)x'') \leq \alpha f(x') + (1-\alpha) f(x'')$$

- The sum of convex functions is a convex function
- Multiplying a convex function by a positive scalar gives a convex function

Convex functions

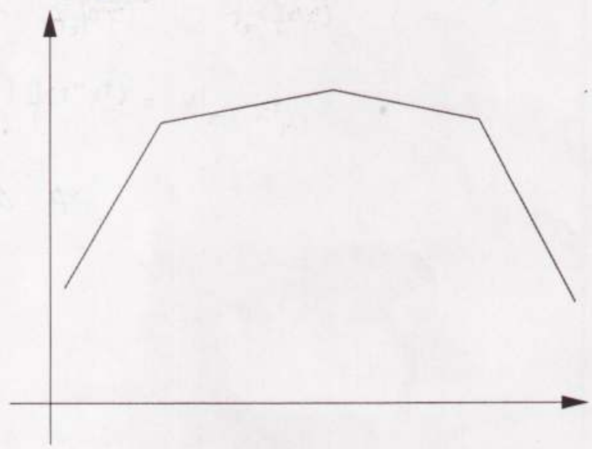
Def.: A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if, for any two points x' and $x'' \in \mathbb{R}^n$ and for any $\alpha \in [0, 1]$

$$f(\alpha x' + (1-\alpha)x'') \leq \alpha f(x') + (1-\alpha) f(x'')$$

- The sum of convex functions is a convex function
- Multiplying a convex function by a positive scalar gives a convex function
- **linear** functions $\sum_{i=1}^{k} a_i x_i$ are convex, irrespective of the sign of a_i's.

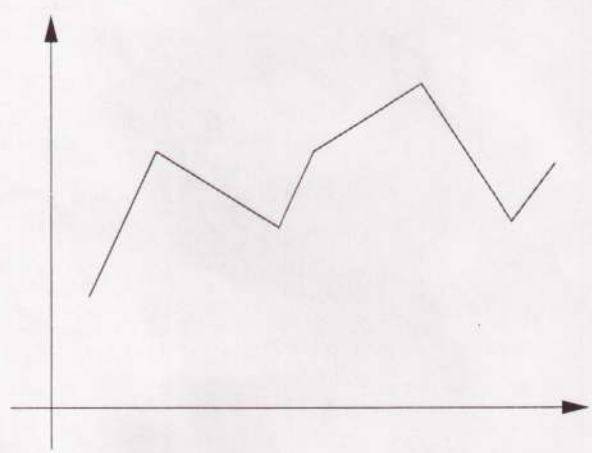
Definition

concave.



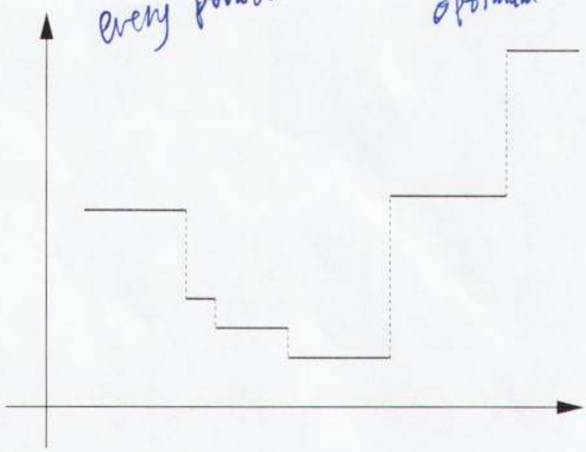
Definition

Definition



Definition

every point is local ~~min~~ optimum



Local and global optima

A vector $x^l \in \mathbb{R}^n$ is a local optimum if

- there is a *neighbourhood*[1] N of x^l with no better point than x^l:

$$\forall x \in N, f_0(x) \geq f_0(x^l)$$

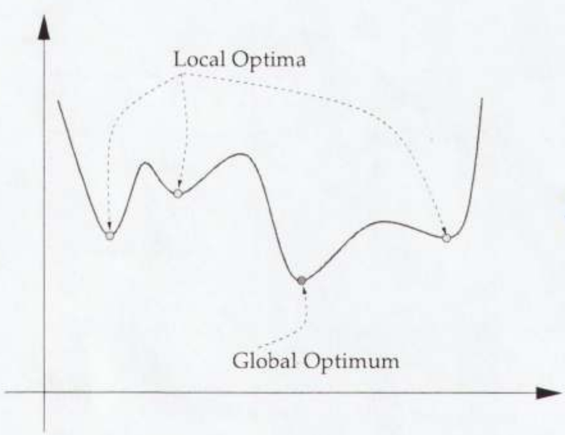
A vector $x^g \in \mathbb{R}^n$ is a global optimum if

- x^g
- there is no x better than x^g, i.e.,

$$f_0(x) \geq f_0(x^g) \qquad \forall x$$

[1] A *neighbourhood* of x^l can be defined as $N = \{x : ||x - x^l||_2 \leq \epsilon\}$ for some ϵ.

Local optima, global optima



Positive (Semi)Definite Matrices

A square $n \times n$ matrix A is Positive Definite (PD) (denoted with $A \succ 0$) if, for any n-vector $x \neq 0$, the following holds:

$$x^\top A x \geq 0$$

$f(x)$. $f''(x) \geq 0$. $\forall x$.

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_ix_j =$$

$$f(x_1,x_2) = x_1^2 + x_2^2 .$$

$$\begin{pmatrix} \frac{\partial^2 f(x_1,x_2)}{\partial x_1^2} & \frac{\partial^2 f(x_1,x_2)}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f(x_1,x_2)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x_1,x_2)}{\partial x_2^2} \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

Positive (Semi)Definite Matrices

A square $n \times n$ matrix A is Positive Definite (PD) (denoted with $A \succ 0$) if, for any n-vector $x \neq 0$, the following holds:

$$x^\top A x \geq 0$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_ix_j =$$

$$\begin{array}{llll} a_{11}x_1^2 & +a_{12}x_1x_2 & +\dots & +a_{1n}x_1x_n + \\ a_{21}x_2x_1 & +a_{22}x_2^2 & +\dots & +a_{2n}x_2x_n + \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}x_nx_1 & +a_{n2}x_nx_1 & +\dots & +a_{nn}x_n^2 \end{array}$$

> 0 PD

≥ 0 PSD.

Some linear algebra

子式.

- A minor of a $m \times n$ matrix A is the determinant of a (square) submatrix of A obtained by removing some rows and columns of A.
- For a square matrix B, a principal minor of B is obtained by removing the same row and column indices from B 主子式.
- For a square matrix B, a leading principal minor of B is a principal minor obtained by removing the last k rows and (the same) columns from B

前主子式.

For example, $A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 6 & 7 \\ 3 & 6 & 8 & 9 \\ 4 & 7 & 9 & 0 \end{pmatrix}$.

Leading principal minors: (1), $\begin{pmatrix} 1 & 2 \\ 2 & 5 \end{pmatrix}$, $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 5 & 6 \\ 3 & 6 & 8 \end{pmatrix}$

Positive (Semi)Definite Hessian Matrices

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) (denote it with $A \succeq 0$) if all principal minors of A are nonnegative.

主子式非负.

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) if and only if $A = BB^\top$ for some $B \in R^{n\times n}$.

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) if for all $i = 1, \ldots, n$ $a_{ii} \geq \sum_{j=1, j\neq i}^{n} |a_{ij}|$. easy

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) if all its eigenvalues are nonnegative.

特征值.

Positive (Semi)Definite Hessian Matrices

Given a function $f : \mathbb{R}^n \to \mathbb{R}$, consider its Hessian:

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

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex on set $\Omega \in R^n$ if and only if the Hessian $H_f(x) \succeq 0$ for all $x \in \Omega$.

Examples

- The function $f(x) = x$ is convex
- The function $f(x_1, x_2) = x_1 + x_2$ is convex
- The function $f(x_1, x_2) = x_1^2 + x_2$ is convex
- The function $f(x_1, x_2) = 5x_1^2 + 3x_2^2$ is convex
- The function $f(x_1, x_2) = x_1^2 + x_2^2 - x_1x_2$ is convex

$\begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$.

3 principle minors.

① (2)

② (0)

③ $\begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$.

$f(x_1, x_2) = x_1^4 + x_2^2 + x_1x_2$

$$\begin{pmatrix} 12x_1^2 & 1 \\ 1 & 2 \end{pmatrix}$$

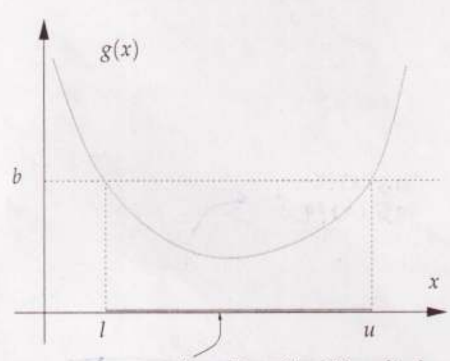
Examples

- The function $f(x) = x$ is convex
- The function $f(x_1, x_2) = x_1 + x_2$ is convex
- The function $f(x_1, x_2) = x_1^2 + x_2$ is convex
- The function $f(x_1, x_2) = 5x_1^2 + 3x_2^2$ is convex
- The function $f(x_1, x_2) = x_1^2 + x_2^2 - x_1x_2$ is convex
- The function $f(x_1, x_2) = x_1^2 + x_2^2 + 5x_1x_2$ is nonconvex
- The function $f(x_1, x_2) = x_1^2 - x_2^2$ is nonconvex
- The function $f(x_1, x_2) = x_1x_2$ is nonconvex
- The function $f(x) = \sin x$, for $x \in [0, 2\pi]$ is nonconvex
- The function $f(x) = -x^2$ is nonconvex

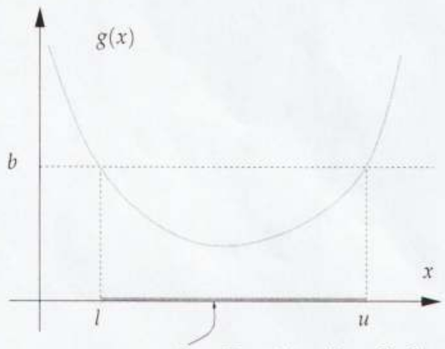
Convex constraints

- A constraint $g(x) \leq b$, with $g : \mathbb{R}^n \to \mathbb{R}$, defines a subset S of $\mathbb{R}^n$, that is,
  $$S = \{x \in \mathbb{R}^n : g(x) \leq b\}$$
- The constraint $g(x) \leq b$ is convex if the set S is convex.
- If the *function* $g(x)$ is convex, the *constraint* $g(x) \leq b$ is convex.

Convex constraints



Convex constraints



$\{x \in \mathbb{R} : g(x) \leq b\} = [l, u]$

Note: if the *function* $g(x)$ is convex, the *constraint* $g(x) \geq b$ may be nonconvex!

$g(x) = x_1^2 + x_2^2$. convex

$g(x) \leq 2$. convex

$g(x) \geq 2$. nonconvex.

Convex and concave functions

Def.: A function $f : \mathbb{R}^n \to \mathbb{R}$ is concave if $-f(x)$ is convex.

- concave functions are useful with maximization problems:

$$
\begin{aligned}
\max \quad & f(x) \\
& g_i(x) \leq 0 \quad \forall i = 1, 2 \ldots, m \\
& h_i(x) = 0 \quad \forall i = m+1, m+2 \ldots, m+q \\
& x \in \mathbb{R}^n
\end{aligned}
\quad \rightarrow \begin{cases} h_i(x) \leq 0 \\ -h_i(x) \leq 0. \end{cases}
$$

is a convex problem if $f(x)$ is concave, all $g_i(x)$ are convex, and all $h_i(x)$ are affine.

- concave functions are also useful for $\geq$ constraints:

$$g_i(x) \geq 0$$

is a convex constraint if $g_i(x)$ is concave.

~~only~~ $\overline{h(x) \text{ is linear}}$. $(\ldots) = 0$.

only if $\begin{cases} h(x) & \text{— convex} \\ -h(x) & \text{— convex} \end{cases}$

For what convex $g(x)$ the *constraint* $g(x) \geq b$ is always convex?

For what convex $g(x)$ the *constraint* $g(x) \geq b$ is always convex?

- linear constraints $\sum_{i=1}^{k} a_i x_i \begin{Bmatrix} \leq \\ = \\ \geq \end{Bmatrix} b$ are convex

The general optimization problem

Consider a vector $x \in \mathbb{R}^n$ of variables.
An optimization problem can be expressed as:

$$
\begin{aligned}
\mathbf{P}: \quad \text{minimize} \quad & f_0(x) \\
\text{s.t.} \quad & f_1(x) \leq b_1 \\
& f_2(x) \leq b_2 \\
& \vdots \\
& f_m(x) \leq b_m
\end{aligned}
$$

Feasible solutions, local and global optima

Define $F = \{x \in \mathbb{R}^n : f_1(x) \leq b_1, f_2(x) \leq b_2, \ldots, f_m(x) \leq b_m\}$, that is, F is the feasible set of an optimization problem.
All points $x \in F$ are called feasible solutions.

A vector $x^l \in \mathbb{R}^n$ is a local optimum if

- $x^l \in F$
- there is a *neighbourhood*[2] N of x^l with no better point than x^l:

$$\forall x \in N \cap F, f_0(x) \geq f_0(x^l)$$

A vector $x^g \in \mathbb{R}^n$ is a global optimum if

- $x^g \in F$
- there is no $x \in F$ better than x^g, i.e.,

$$f_0(x) \geq f_0(x^g) \qquad \forall x \in F$$

[2] A *neighbourhood* of x^l can be defined as $N = \{x : ||x - x^l||_2 \leq \epsilon\}$ for some ϵ.

Convex problems

Def.: An optimization problem is convex if

- the objective function is convex
- all constraints are convex

Convex problems

Def.: An optimization problem is convex if

- the objective function is convex
- all constraints are convex

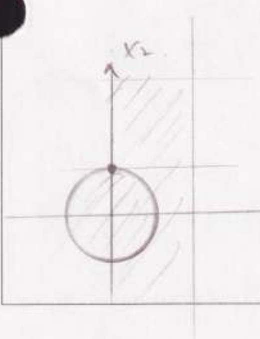
Convex optimization problems are *easy*

> **If a problem P is convex, a local optimum x^* of P is also a global optimum of P.**

(Hint) When modeling an optimization problem, it would be good if we found a convex problem.

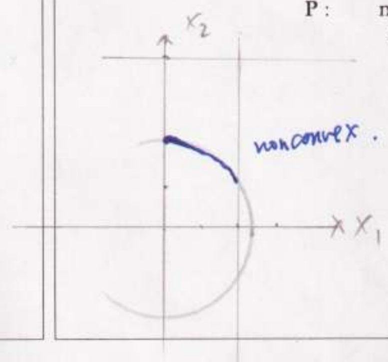
Examples of convex problems

$$\begin{array}{lll} \mathbf{P}: & \min & x_1^2 + 2x_2^2 \\ & \text{s.t.} & x_1^2 + x_2^2 \le 1 \\ & & 0 \le x_1 \le 2 \\ & & 1 \le x_2 \le 5 \end{array}$$



Examples of nonconvex problems

$$\begin{array}{lll} \mathbf{P}: & \min & x_1^2 + 2x_2^2 \\ & \text{s.t.} & x_1^2 + x_2^2 = 5 \\ & & 0 \le x_1 \le 2 \\ & & 1 \le x_2 \le 5 \end{array}$$



Examples of nonconvex problems

$$\begin{array}{lll} \mathbf{P}: & \min & x_1^2 + 2x_2^2 \\ & \text{s.t.} & x_1^2 + x_2^2 \le 5 \\ & & 0 \le x_1 \le 2 \\ & & 1 \le x_2 \le 5 \\ & & x_2 \in \mathbb{Z} \end{array}$$

Examples of "hidden" convex problems

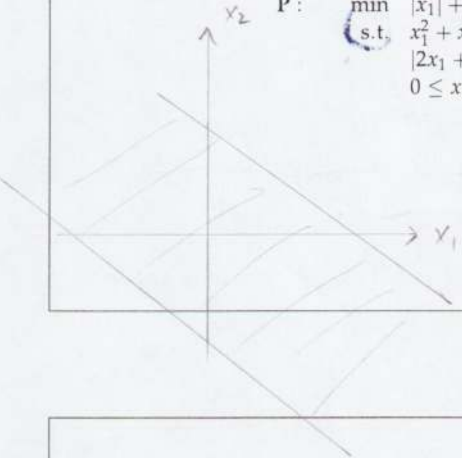
$$\begin{array}{lll} \mathbf{P}: & \min & x_1 - 2x_2^2 \\ & \text{s.t.} & x_1^2 + x_2^2 \le 1 \\ & & x_2 = 0 \\ & & 0 \le x_1 \le 5 \end{array}$$

linear function.

if $f(x) = 0$
↳ is a linear function.
it can be convex. and ~~constraint~~ concave.

Additional examples of convex problems and functions

$$\mathbf{P}: \quad \begin{array}{ll} \min & |x_1| + |x_2| \\ \text{s.t.} & x_1^2 + x_2^2 \leq 1 \\ & |2x_1 + 3x_2| \leq 10 \\ & 0 \leq x_1 \leq 5 \end{array}$$



Your first Optimization model

Variables	r: radius of the can's base
	h: height of the can
Objective	$2\pi rh + 2\pi r^2$ (minimize)
Constraints	$\pi r^2 h = V$
	$h > 0$
	$r > 0$

Relaxations

Relaxation of an Optimization problem

Consider an optimization problem

$$\mathbf{P}: \quad \begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_1(x) \leq b_1 \\ & f_2(x) \leq b_2 \\ & \vdots \\ & f_m(x) \leq b_m, \end{array}$$

Let F denote the set of points x that satisfy all constraints:

$$F = \{x \in \mathbb{R}^n : \begin{array}{l} f_1(x) \leq b_1, \\ f_2(x) \leq b_2, \\ \vdots \\ f_m(x) \leq b_m\} \end{array}$$

So we can write $\mathbf{P} : \min\{f_0(x) : x \in F\}$ for short.

Relaxation of an Optimization problem

Consider a problem $\mathbf{P} : \min\{f_0(x) : x \in F\}$.

A problem $\mathbf{P}' : \min\{f_0'(x) : x \in F'\}$ is a **relaxation** of $\mathbf{P}$ if:

- $F' \supseteq F$
- $f_0'(x) \leq f_0(x)$ for all $x \in F$.[3]

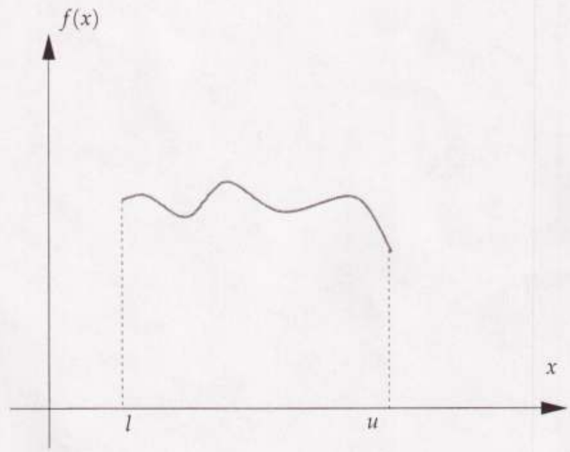
If $\mathbf{P}'$ is a relaxation of a problem $\mathbf{P}$, then the global optimum of $\mathbf{P}'$ is $\leq$ the global optimum of $\mathbf{P}$.

[3] We don't care what $f_0'(x)$ is outside of F.

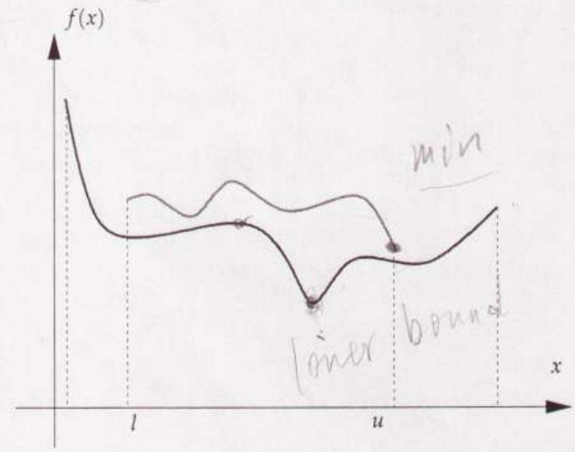
Examples

- $\min\{f(x) : -1 \leq x \leq 1\}$ is a relaxation of $\min\{f(x) : x = 0\}$
- $\min\{f(x) : -1 \leq x \leq 1\}$ is a r. of $\min\{f(x) : 0 \leq x \leq 1\}$
- $\min\{f(x) : -1 \leq x \leq 1\}$ is not a r. of $\min\{f(x) : -2 \leq x \leq 1\}$
- $\min\{f(x) : g(x) \leq b\}$ is a r. of $\min\{f(x) : g(x) \leq b - 1\}$
- $\min\{f(x) - 1 : g(x) \leq b\}$ is a r. of $\min\{f(x) : g(x) \leq b\}$

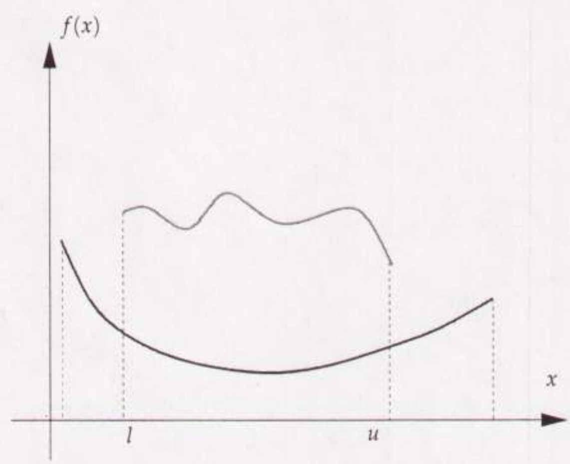
Relaxation of an Optimization problem



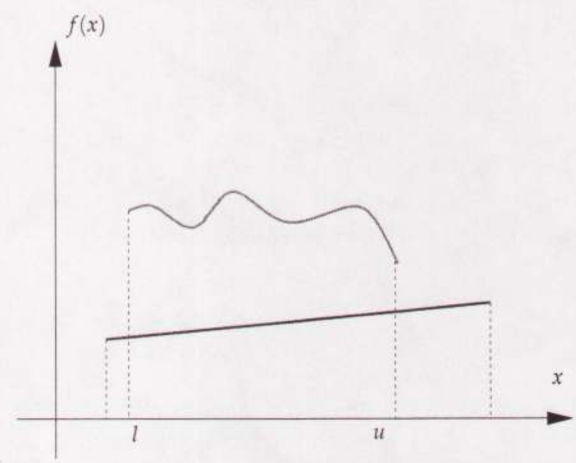
Relaxation of an Optimization problem



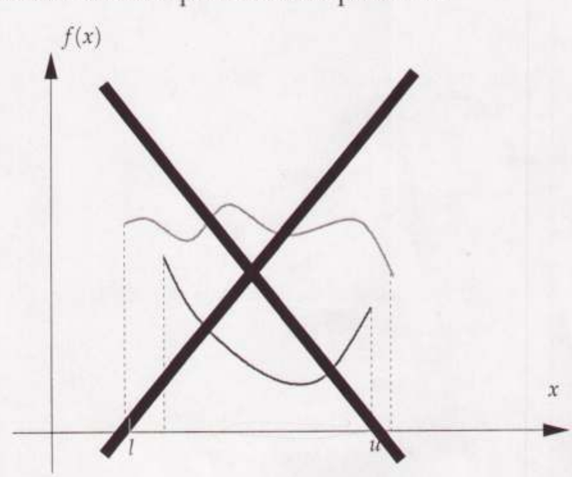
Relaxation of an Optimization problem



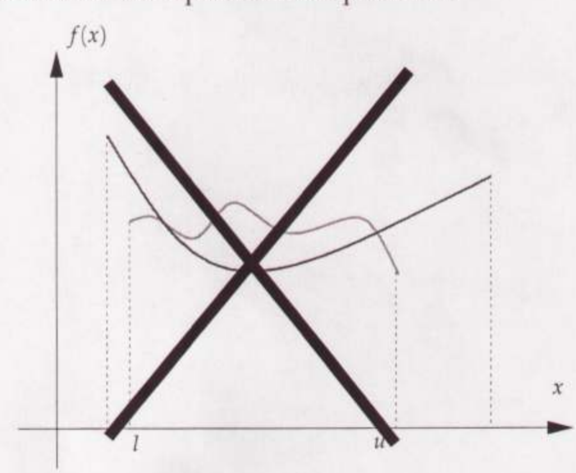
Relaxation of an Optimization problem



Relaxation of an Optimization problem



Relaxation of an Optimization problem



Relaxation of an Optimization problem

Consider again a problem
$\mathbf{P} : \min\{f_0(x) : f_1(x) \leq b_1, f_2(x) \leq b_1, \ldots, f_m(x) \leq b_m\}$, or
$\mathbf{P} : \min\{f_0(x) : x \in F\}$ for short.

- deleting a constraint from **P** provides a relaxation of **P**.
- adding a constraint $f_{m+1}(x) \leq b_{m+1}$ to a problem **P** does the opposite:

$$F'' = \{x \in \mathbb{R}^n : \begin{array}{l} f_1(x) \leq b_1, \\ f_2(x) \leq b_2, \\ \ldots, \\ f_m(x) \leq b_m, \\ f_{m+1}(x) \leq b_{m+1}\} \subseteq F \end{array}$$

and therefore

$$\min\{f_0(x) : x \in F''\} \quad \geq \quad \min\{f_0(x) : x \in F\}$$

Upper & Lower bounds

Lower and upper bounds

Consider an optimization problem $\mathbf{P} : \min\{f_0(x) : x \in F\}$:

- for any feasible solution $x \in F$, the corresponding objective function value $f_0(x)$ is an upper bound.
- the most interesting upper bounds are the local optima.
- a lower bound of **P** is instead a value z such that

$$z \leq \min\{f_0(x) : x \in F\}.$$

Upper vs. Lower bounds

Situation #1:

You: "We found a solution that will only cost 372,000 \$."
Boss: "Ok, that sounds good."

Upper vs. Lower bounds

Situation #1:

You: "We found a solution that will only cost 372,000 \$."
Boss: "Ok, that sounds good."

Situation #2:

You: "We found a solution that will only cost 372,000 \$."
Boss: "That's too much, find something better."

Upper vs. Lower bounds

Situation #1:

You: "We found a solution that will only cost 372,000 \$."
Boss: "Ok, that sounds good."

Situation #2:

You: "We found a solution that will only cost 372,000 \$."
Boss: "That's too much, find something better."

...

You: "We found another solution that costs 354,000 \$."
Boss: "Can't you do better than that?"

Upper vs. Lower bounds

Situation #1:

You: "We found a solution that will only cost 372,000 \$."

Boss: "Ok, that sounds good."

Situation #2:

You: "We found a solution that will only cost 372,000 \$."

Boss: "That's too much, find something better."

...

You: "We found another solution that costs 354,000 \$."

Boss: "Can't you do better than that?"

You: "I can try again, but here's the proof that we can't go below 351,500."

Boss: "Ok then, that's a good solution."

What relaxations are for

- If $\mathbf{P}'$ is a relaxation of a problem $\mathbf{P}$, then the global optimum of $\mathbf{P}'$ is $\leq$ the global optimum of $\mathbf{P}$.
- Hence, any relaxation $\mathbf{P}'$ of $\mathbf{P}$ provides a lower bound on $\mathbf{P}$.

⇒ If a problem $\mathbf{P}$ is difficult but a relaxation $\mathbf{P}'$ of $\mathbf{P}$ is easier to solve than $\mathbf{P}$ itself, we can still try and solve $\mathbf{P}'$: (i) we get a lower bound and (ii) the solution of $\mathbf{P}'$ may help solve $\mathbf{P}$.

To recap: the Knapsack problem

At a flea market in Rome, you spot n objects (old pictures, a vessel, rusty medals...) that you could re-sell in your antique shop for about double the price.

- You want these objects to pay for your flight ticket to Rome, which cost C.
- Also, you don't want a heavy backpack, so you want to buy the objects that will load it as little as possible.

How do you solve this problem?

The Knapsack problem

Each object $i = 1, 2, \dots, n$ has a price p_i and a weight w_i.

- Variables: one variable x_i for each $i = 1, 2 \dots, n$. This is a "yes/no" variable: either you take the i-th object or not.

$$x_i = \begin{cases} 1, & \text{if buy} \\ 0, & \text{if not.} \end{cases}$$

The Knapsack problem

Each object $i = 1, 2, \dots, n$ has a price p_i and a weight w_i.

- Variables: one variable x_i for each $i = 1, 2 \dots, n$. This is a "yes/no" variable: either you take the i-th object or not.
- Constraint: total revenue must be at least C

The Knapsack problem

Each object $i = 1, 2, \dots, n$ has a price p_i and a weight w_i.

- Variables: one variable x_i for each $i = 1, 2 \dots, n$. This is a "yes/no" variable: either you take the i-th object or not.
- Constraint: total revenue must be at least C

  (As you'll double the price when selling them at your store, the revenue for each object is exactly p_i)

The Knapsack problem

Each object $i = 1, 2, \ldots, n$ has a price p_i and a weight w_i.

- Variables: one variable x_i for each $i = 1, 2 \ldots, n$. This is a "yes/no" variable: either you take the i-th object or not.
- Constraint: total revenue must be at least C
  (As you'll double the price when selling them at your store, the revenue for each object is exactly p_i)
- Objective function: the total weight

Your first (non-trivial) optimization model

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & \textstyle\sum_{i=1}^n p_i x_i \geq C \\ & x_i \in \{0,1\} \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

Is it convex?

No.

Relaxing the Knapsack problem

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & x_i \in \{0,1\} \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

$x_i^2 - x_i = 0$

Relaxing the Knapsack problem

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & x_i \in \{0,1\} \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

This relaxation would give us $x_i = 0$ for all $i = 1, 2, \ldots, n$, and a lower bound of $\sum_{i=1}^n w_i x_i = 0$. Not so great...

Relaxing the Knapsack problem

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & x_i \in \{0,1\} \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

This relaxation would give us $x_i = 0$ for all $i = 1, 2, \ldots, n$, and a lower bound of $\sum_{i=1}^n w_i x_i = 0$. Not so great...

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & \textstyle\sum_{i=1}^n p_i x_i \geq C \\ & 0 \leq x_i \leq 1 \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

Relaxing the Knapsack problem

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & x_i \in \{0,1\} \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

This relaxation would give us $x_i = 0$ for all $i = 1, 2, \ldots, n$, and a lower bound of $\sum_{i=1}^n w_i x_i = 0$. Not so great...

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^n w_i x_i \\ & \textstyle\sum_{i=1}^n p_i x_i \geq C \\ & 0 \leq x_i \leq 1 \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

- Relaxing integrality of the variables gives a relaxation where we admit fractions of objects.

$\approx$ we pulverized all objects and took some spoonfuls of each

- It doesn't make sense, but it's a relaxation, and it **does** give us a better lower bound.

ISE 426
Optimization models and applications

Lecture 3 — September 3, 2015

- Upper & lower bounds
- Relaxations: an example
- Linear programming

What relaxations are for

- If $\mathbf{P}'$ is a relaxation of a problem $\mathbf{P}$, then the global optimum of $\mathbf{P}'$ is $\leq$ the global optimum of $\mathbf{P}$.
- Hence, any relaxation $\mathbf{P}'$ of $\mathbf{P}$ provides a lower bound on $\mathbf{P}$.

$\Rightarrow$ If a problem $\mathbf{P}$ is difficult but a relaxation $\mathbf{P}'$ of $\mathbf{P}$ is easier to solve than $\mathbf{P}$ itself, we can still try and solve $\mathbf{P}'$: (i) we get a lower bound and (ii) the solution of $\mathbf{P}'$ may help solve $\mathbf{P}$.

The Knapsack problem

At a flea market in Rome, you spot n objects (old pictures, a vessel, rusty medals...) that you could re-sell in your antique shop for about double the price.

- You want these objects to pay for your flight ticket to Rome, which cost C.
- Also, your backpack can carry all of them, but you don't want it heavy, so you want to buy the objects that will load your backpack as little as possible.

How do you solve this problem?

The Knapsack problem

Each object $i = 1, 2 \ldots, n$ has a price $p_i > 0$ and a weight $w_i > 0$.

- Variables: one variable x_i for each $i = 1, 2 \ldots, n$.

$\Rightarrow$ x_i is a "yes/no" variable: either you take the i-th object ($x_i = 1$) or you do not ($x_i = 0$).

- Constraint: total revenue must be at least C

  (As you'll double the price when selling them at your store, the revenue for each object is exactly p_i)
- Objective function: the total weight

Your first (non-trivial) optimization model

$$\begin{array}{ll} \mathbf{P}: \min & \sum_{i=1}^n w_i x_i \\ & \sum_{i=1}^n p_i x_i \geq C \\ & x_i \in \{0,1\} \quad \forall i = 1,2,\ldots,n \end{array}$$

Nonconvex!

Your first (non-trivial) optimization model

$$\begin{array}{ll} \mathbf{P}: \min & \sum_{i=1}^n w_i x_i \\ & \sum_{i=1}^n p_i x_i \geq C \\ & x_i \in \{0,1\} \quad \forall i = 1,2,\ldots,n \end{array}$$

Nonconvex! Relaxation #1:

$$\begin{array}{ll} \mathbf{R1}: \min & \sum_{i=1}^n w_i x_i \\ & x_i \in \{0,1\} \quad \forall i = 1,2,\ldots,n \end{array}$$

This relaxation gives us $x_i = 0$ for all $i = 1, 2, \ldots, n$, and a lower bound of $\sum_{i=1}^n w_i x_i = 0$. Not so great...

Your first (non-trivial) optimization model

$$\mathbf{P}: \min \sum_{i=1}^{n} w_i x_i \quad \sum_{i=1}^{n} p_i x_i \geq C \quad x_i \in \{0,1\} \quad \forall i = 1,2,\ldots,n$$

Nonconvex! Relaxation #1:

$$\mathbf{R1}: \min \sum_{i=1}^{n} w_i x_i \quad x_i \in \{0,1\} \quad \forall i = 1,2,\ldots,n$$

This relaxation gives us $x_i = 0$ for all $i = 1,2,\ldots,n$, and a lower bound of $\sum_{i=1}^{n} w_i x_i = 0$. Not so great... Relaxation #2:

$$\mathbf{R2}: \min \sum_{i=1}^{n} w_i x_i \quad \sum_{i=1}^{n} p_i x_i \geq C \quad 0 \leq x_i \leq 1 \quad \forall i = 1,2,\ldots,n$$

. [illegible]

By relaxing integrality we admit fractions of objects.
It is as if we pulverized object and took some spoonful of each.
Nonsense? It's a relaxation, and it gives a lower bound.

Example

Suppose there are $n = 9$ objects and $C = 70$.

i	1	2	3	4	5	6	7	8	9
p_i	27	24	8	35	29	8	31	18	12
w_i	3	2	2	4	5	4	3	1	4

Example

Suppose there are $n = 9$ objects and $C = 70$.

i	1	2	3	4	5	6	7	8	9
p_i	27	24	8	35	29	8	31	18	12
w_i	3	2	2	4	5	4	3	1	4

9 4

A local optimum is 8, solution is $(1,1,1,0,0,0,0,1,0)$.

$70 - 42 = 28$ ·

$(0,1,1,0,0,0,1,1,0)$

min W.

Example

Suppose there are $n = 9$ objects and $C = 70$.

i	1	2	3	4	5	6	7	8	9
p_i	27	24	8	35	29	8	31	18	12
w_i	3	2	2	4	5	4	3	1	4

A local optimum is 8, solution is $(1,1,1,0,0,0,0,1,0)$.

R#1: lower bound is 0, solution is $(0,0,0,0,0,0,0,0,0)$.

$x_8 = 1$

$x_2 = 1$

$x_7 = \frac{28}{31}$

Example

Suppose there are $n = 9$ objects and $C = 70$.

i	1	2	3	4	5	6	7	8	9
p_i	27	24	8	35	29	8	31	18	12
w_i	3	2	2	4	5	4	3	1	4

A local optimum is 8, solution is $(1,1,1,0,0,0,0,1,0)$.

R#1: lower bound is 0, solution is $(0,0,0,0,0,0,0,0,0)$.

R#2: lower bound is 5.71, solution is $(0,1,0,0,0,0.903,1,0)$.

Now, I know 6 is the best.
I can do

Example

Suppose there are $n = 9$ objects and $C = 70$.

i	1	2	3	4	5	6	7	8	9
p_i	27	24	8	35	29	8	31	18	12
w_i	3	2	2	4	5	4	3	1	4

A local optimum is 8, solution is $(1,1,1,0,0,0,0,1,0)$.

R#1: lower bound is 0, solution is $(0,0,0,0,0,0,0,0,0)$.

R#2: lower bound is 5.71, solution is $(0,1,0,0,0,0.903,1,0)$.

Global optimum is 6, solution is $(0,1,0,0,0,0,1,1,0)$.

To recap

- convex problems are good
- if model is nonconvex, look for a (possibly convex) relaxation
- use it to get a lower bound!

Linear Programming

Linear programming

Consider the optimization problem:

$$\mathbf{P}: \quad \min \sum_{i=1}^{n} c_i x_i \\ \sum_{i=1}^{n} a_{ji} x_i \geq b_j \quad \forall j = 1, 2, \ldots, m \\ l_i \leq x_i \leq u_i \quad \forall i = 1, 2, \ldots, n,$$

with n variables and $m + n$ constraints. Problems like **P** are called **Linear Programming** (LP) problems.
They are often written in matricial form:

$$\mathbf{P}: \quad \min c^T x \\ Ax \geq b \\ l \leq x \leq u$$

A is the *coefficient matrix*,

Linear programming

Consider the optimization problem:

$$\mathbf{P}: \quad \min \sum_{i=1}^{n} c_i x_i \\ \sum_{i=1}^{n} a_{ji} x_i \geq b_j \quad \forall j = 1, 2, \ldots, m \\ l_i \leq x_i \leq u_i \quad \forall i = 1, 2, \ldots, n,$$

with n variables and $m + n$ constraints. Problems like **P** are called **Linear Programming** (LP) problems.
They are often written in matricial form:

$$\mathbf{P}: \quad \min c^T x \\ Ax \geq b \\ l \leq x \leq u$$

A is the *coefficient matrix*, b is the *right-hand side vector*, and c is the *objective coefficient vector*. We call l_i and u_i *lower* and *upper* bound on variable x_i. They don't need to be finite.

LP problems are convex, therefore they are "easy".

Example

You are cast on an island in the middle of the Pacific, and the only source of food is a well-known restaurant.

Example

You are cast on an island in the middle of the Pacific, and the only source of food is a well-known restaurant. Here's the menu:

QP:	Quarter Pounder	FR:	Fries (small)
MD:	McLean Deluxe	SM:	Sausage McMuffin
BM:	Big Mac	1M:	1% Lowfat Milk
FF:	Filet-O-Fish	OJ:	Orange Juice
MC:	McGrilled Chicken		

Each food has a different combination of nutrient (proteins, Vitamin A, Iron, etc.) and a cost. You want to

- get the necessary nutrients every day (constraint!)
- minimize the total cost of the foods (objective function)
- what are the variables?

Nutrients

	QP	MD	BM	FF	MC	FR	SM	1M	OJ	
Cost	1.84	2.19	1.84	1.44	2.29	0.77	1.29	0.60	0.72	Req'd
Prot	28	24	25	14	31	3	15	9	1	55
VitA	15%	15%	6%	2%	8%	0%	4%	10%	2%	100%
VitC	6%	10%	2%	0%	15%	15%	0%	4%	120%	100%
Calc	30%	20%	25%	15%	15%	0%	20%	30%	2%	100%
Iron	20%	20%	20%	10%	8%	2%	15%	0%	2%	100%
Cals	510	370	500	370	400	220	345	110	80	2000
Carb	34	35	42	38	42	26	27	12	20	350

Model

Define $F = \{QP, MD, BM, FF, MC, FR, SM, 1M, OJ\}$

Model

Define $F = \{QP, MD, BM, FF, MC, FR, SM, 1M, OJ\}$ and $N = \{\text{Prot}, \text{VitA}, \text{VitC}, \text{Calc}, \text{Iron}, \text{Cals}, \text{Carb}\}$.

- define variable x_i as the amount of food i you will buy every day ($i \in F$)

Model

Define $F = \{QP, MD, BM, FF, MC, FR, SM, 1M, OJ\}$ and $N = \{\text{Prot}, \text{VitA}, \text{VitC}, \text{Calc}, \text{Iron}, \text{Cals}, \text{Carb}\}$.

- define variable x_i as the amount of food i you will buy every day ($i \in F$)
- define parameters:
  - c_i is the cost per unit of food i
  - a_{ij} is the amount of nutrient $j \in N$ per unit of food $i \in F$
  - b_j is the amount of nutrient $j \in N$ required every day

Then the optimization model is an LP model:

$$\begin{array}{ll} \min & c^T x \\ & Ax \geq b \\ & l \leq x \leq u \end{array}$$

Overall model

$$\begin{array}{lrrrrrrrrrl}
\min & 1.84x_{qp} & +2.19x_{md} & +1.84x_{bm} & +1.44x_{ff} & +2.29x_{mc} & +0.77x_{fr} & +1.29x_{sm} & +0.60x_{1m} & +0.72x_{oj} & \\
(Prot) & 28x_{qp} & +24x_{md} & +25x_{bm} & +14x_{ff} & +31x_{mc} & +3x_{fr} & +15x_{sm} & +9x_{1m} & +1x_{oj} & \geq 55 \\
(VitA) & 15x_{qp} & +15x_{md} & +6x_{bm} & +2x_{ff} & +8x_{mc} & & +4x_{sm} & +10x_{1m} & +2x_{oj} & \geq 100 \\
(VitC) & 6x_{qp} & +10x_{md} & +2x_{bm} & & +15x_{mc} & +15x_{fr} & & +4x_{1m} & +120x_{oj} & \geq 100 \\
(Calc) & 30x_{qp} & +20x_{md} & +25x_{bm} & +15x_{ff} & +15x_{mc} & & +20x_{sm} & +30x_{1m} & +2x_{oj} & \geq 100 \\
(Iron) & 20x_{qp} & +20x_{md} & +20x_{bm} & +10x_{ff} & +8x_{mc} & +2x_{fr} & +15x_{sm} & & +2x_{oj} & \geq 100 \\
(Cals) & 510x_{qp} & +370x_{md} & +500x_{bm} & +370x_{ff} & +400x_{mc} & +220x_{fr} & +345x_{sm} & +110x_{1m} & +80x_{oj} & \geq 2000 \\
(Carb) & 34x_{qp} & +35x_{md} & +42x_{bm} & +38x_{ff} & +42x_{mc} & +26x_{fr} & +27x_{sm} & +12x_{1m} & +20x_{oj} & \geq 350
\end{array}$$

Nonnegativity constraint: $x_i \geq 0, i \in F$.

Another example

The manager of a post office is hiring new employees

They can be full or part time. The part time can be 1% to 99% — this is only to simplify the problem. Rules:

- at least these many employees each day of the week:

day	S	M	T	W	Th	F	Sa
# empl.	11	17	13	15	19	14	16

- (state regulations impose) that an employee works five days in a row and then receives two days off
- that the number of employees is minimum

What are the **variables** of the problem?

Another example

The manager of a post office is hiring new employees

They can be full or part time. The part time can be 1% to 99% — this is only to simplify the problem. Rules:

- at least these many employees each day of the week:

day	S	M	T	W	Th	F	Sa
# empl.	11	17	13	15	19	14	16

- (state regulations impose) that an employee works five days in a row and then receives two days off
- that the number of employees is minimum

What are the **variables** of the problem?

- the number of employees working each day?
- the total number of employees to hire?

What do I (as a boss) want to know at the end?

What to we want to know?

- If an employee works on Thu, his/her work days can be
  - Thu, Fri, Sat, Sun, Mon, or
  - Wed, Thu, Fri, Sat, Sun, or
  - Tue, Wed, Thu, Fri, Sat, or
  - Mon, Tue, Wed, Thu, Fri, or
  - Sun, Mon, Tue, Wed, Thu.

⇒ We don't know when he/she started his working shift.

- It is the variable we are looking for!
- Actually, we are only interested in...

What to we want to know?

- If an employee works on Thu, his/her work days can be
  - Thu, Fri, Sat, Sun, Mon, or
  - Wed, Thu, Fri, Sat, Sun, or
  - Tue, Wed, Thu, Fri, Sat, or
  - Mon, Tue, Wed, Thu, Fri, or
  - Sun, Mon, Tue, Wed, Thu.

⇒ We don't know when he/she started his working shift.

- It is the variable we are looking for!
- Actually, we are only interested in...

  the number of employees starting **on a certain day**
- Define it as variable x_i, with $i \in \{\text{Sun}, \text{Mon}, \text{Tue}, \text{Wed}, \text{Thu}, \text{Fri}, \text{Sat}\}$.

Now that we know what we are looking for...

We have variables. We can write constraints & objective f.

- constraint #1: there must be 19 employees on Thursdays.

Now that we know what we are looking for...

We have variables. We can write constraints & objective f.

- constraint #1: there must be 19 employees on Thursdays.

$$x_{Thu} + x_{Wed} + x_{Tue} + x_{Mon} + x_{Sun} \geq 19$$

- constraint #2: an employee works five consecutive days and then receives two days off.

Now that we know what we are looking for...

We have variables. We can write constraints & objective f.

- constraint #1: there must be 19 employees on Thursdays.

$$x_{Thu} + x_{Wed} + x_{Tue} + x_{Mon} + x_{Sun} \geq 19$$

- constraint #2: an employee works five consecutive days and then receives two days off. This is already included in the definition of our variables **and** in the above constraint.

Now that we know what we are looking for...

We have variables. We can write constraints & objective f.

- constraint #1: there must be 19 employees on Thursdays.

$$x_{Thu} + x_{Wed} + x_{Tue} + x_{Mon} + x_{Sun} \geq 19$$

- constraint #2: an employee works five consecutive days and then receives two days off. This is already included in the definition of our variables **and** in the above constraint.
- objective function: the total number of employees (to be minimized).

Now that we know what we are looking for...

We have variables. We can write constraints & objective f.

- constraint #1: there must be 19 employees on Thursdays.

$$x_{Thu} + x_{Wed} + x_{Tue} + x_{Mon} + x_{Sun} \geq 19$$

- constraint #2: an employee works five consecutive days and then receives two days off. This is already included in the definition of our variables **and** in the above constraint.
- objective function: the total number of employees (to be minimized).

⇒ number of employees starting on Monday, plus those starting on Tuesday, etc.

- we can sum them up because they define disjoint sets of employees: if one starts working on Thursday, he doesn't start on Friday...

The model

$$\begin{array}{llllllllll}
\min & x_{Sun} & +x_{Mon} & +x_{Tue} & +x_{Wed} & +x_{Thu} & +x_{Fri} & +x_{Sat} & \\
(Sun) & x_{Sun} & & & +x_{Wed} & +x_{Thu} & +x_{Fri} & +x_{Sat} & \geq 11 \\
(Mon) & x_{Sun} & +x_{Mon} & & & +x_{Thu} & +x_{Fri} & +x_{Sat} & \geq 17 \\
(Tue) & x_{Sun} & +x_{Mon} & +x_{Tue} & & & +x_{Fri} & +x_{Sat} & \geq 13 \\
(Wed) & x_{Sun} & +x_{Mon} & +x_{Tue} & +x_{Wed} & & & +x_{Sat} & \geq 15 \\
(Thu) & x_{Sun} & +x_{Mon} & +x_{Tue} & +x_{Wed} & +x_{Thu} & & & \geq 19 \\
(Fri) & & x_{Mon} & +x_{Tue} & +x_{Wed} & +x_{Thu} & +x_{Fri} & & \geq 14 \\
(Sat) & & & x_{Tue} & +x_{Wed} & +x_{Thu} & +x_{Fri} & +x_{Sat} & \geq 16 \\
 & x_{Sun}, & x_{Mon}, & x_{Tue}, & x_{Wed}, & x_{Thu}, & x_{Fri}, & x_{Sat} & \geq 0
\end{array}$$

The solution

LP: with part-time contracts (here $\frac{1}{3}$-time contracts used).

IP: solution with only full-time contracts.

	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Total
LP	5	$1+\frac{1}{3}$	$5+\frac{1}{3}$	0	$7+\frac{1}{3}$	0	$3+\frac{1}{3}$	$22+\frac{1}{3}$
IP	4	1	6	0	8	0	4	23

ISE 426
Optimization models and applications

Lecture 4 — September 8, 2015

- Linear programming: more examples
- Basic properties of LP problems

Reading: WV p. 56.

LP example #3: Short term financial planning[1]

- Your company makes tape recorders (TR) and radios (RD).
- Returns are:
  TR: 100$ (price) − 50$ (labor) − 30$ (raw m.)= 20$
  RD: 90$ (price)− 35$ (labor) − 40$ (raw m.) = 15$
- Raw material is sufficient to produce 100 TRs and 100 RDs

[1] See Winston & Venkataramanan, page 82.

LP example #3: Short term financial planning

Balance sheet (today):	Assets	Liabilities
Cash	\$10,000	
Accts. recv.	\$3,000	
Inv. outst.	\$7,000[2]	
Bank loan		\$10,000

Before the end of the month,

- we will collect soon \$2,000 of accts.
- we will receive new inventory worth \$2000
- we must pay \$1,000 of loan and another \$1,000 for rental
- management: "on 09/30 cash has to be at least \$4,000"
- bank requires that assets / liability ratio be at least 2

⇒ How many TRs and RDs do we produce this month to maximize return?

[2] $\$7000 = \$30 \times 100 + \$40 \times 100$.

LP example #3: Short term financial planning

- return on each TR is \$20, RD is \$15
- suppose t is #TR and r is #RD
- Balance sheet in a month:

BS (09/30):	Assets	Liabilities
Cash	\$10,000	
	$+\$2{,}000 - \$1{,}000 - \$1{,}000$	
	$-\$50t - \$35r$	
Accts. recv.	\$3,000	
	$-\$2{,}000 + \$100t + \$90r$	
Inv. outst.	\$7,000	
	$+\$2{,}000 - \$30t - \$40r$	
Bank loan		\$10,000
		$+\$2{,}000 - \$1{,}000$

LP example #3: Short term financial planning

- return on each TR is \$20, RD is \$15
- suppose t is #TR and r is #RD
- Balance sheet in a month:

BS (09/30):	Assets	Liabilities
Cash	$\$10{,}000 - \$50t - \$35r$	
Accts. recv.	$\$1{,}000 + \$100t + \$90r$	
Inv. outst.	$\$9{,}000 - \$30t - \$40r$	
Bank loan		\$11,000

- Cash $\geq \$4,000$ means $\$10,000 - \$50t - \$35r \geq \$4,000$

$\Rightarrow \$50t + \$35r \leq \$6,000$

- Ratio ≥ 2 means $\dfrac{\text{Cash + Accts. recv. + Inv. outst.}}{\text{Bank loan}} \geq 2$

$\Rightarrow \dfrac{\$20,000 + \$20t + \$15r}{\$11,000} \geq 2$

$\Rightarrow \$20,000 + \$20t + \$15r \geq \$22,000$

$\Rightarrow \$20t + \$15r \geq \$2,000$

LP example #3: Short term financial planning

$$\begin{array}{llll} \max & \$20t & +15r & \\ & \$50t & +\$35r & \leq \$6,000 \\ & \$20t & +\$15r & \geq \$2,000 \\ & t & & \leq 100 \\ & & r & \leq 100 \\ & t, & r & \geq 0 \end{array}$$

LP example #4: Project selection[4]

We have 5 investment opportunities over a 2-year term.

- i.e., we'll invest in the same funds this and next year
- each has two cash outflows, for 2009 and for 2010, and
- a Net Present Value (NPV)[3]
- available cash: 40 M$ this year, estimate 20 M$ next year

Investment		1	2	3	4	5
(a_i) Cash outflow,	2009	11	53	5	5	29
(b_i)	2010	3	6	5	1	34
(v_i) NPV		13	16	16	14	39

What investment(s) get the **maximum** total NPV? What percentage of each?

[3] The amount by which the investment will increase the company's value.
[4] Winston&Venkataramanan, example 10, page 80.

LP example #4: Project selection

- Variables: for each opportunity 1, 2,…, 5, the percentage of investment: $x_i \in [0,1] \forall i = 1, 2, \ldots, 5$
- Constraints: limited cash to expend in 2009 and in 2010:

$$\sum_{i=1}^{5} a_i x_i \leq 40 \qquad \sum_{i=1}^{5} b_i x_i \leq 20$$

- Objective function: the total NPV (to be maximized)

$$\sum_{i=1}^{5} v_i x_i$$

LP example #4: Project selection

$$\begin{array}{lrrrrrl}
\max & 13x_1 & +16x_2 & +16x_3 & +14x_4 & +39x_5 & \\
 & 11x_1 & +53x_2 & +5x_3 & +5x_4 & +29x_5 & \leq 40 \\
 & 3x_1 & +6x_2 & +5x_3 & +1x_4 & +34x_5 & \leq 20 \\
 & x_1, & x_2, & x_3, & x_4, & x_5 & \in [0,1]
\end{array}$$

Graphical solution of LP problems

Consider an LP problem with m constraints and **two variables**.

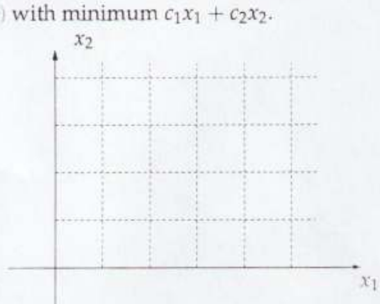
$$\begin{array}{llll}
\min & c_1x_1 & +c_2x_2 & \\
 & a_{11}x_1 & +a_{12}x_2 & \leq b_1 \\
 & a_{21}x_1 & +a_{22}x_2 & \leq b_2 \\
 & \vdots & & \\
 & a_{m1}x_1 & +a_{m2}x_2 & \leq b_m
\end{array}$$

- the objective function is associated with vector (c_1, c_2) in $\mathbb{R}^2$
- lines defined by $c_1x_1 + c_2x_2 = c_0$ correspond to solutions with the same objective function, c_0
- "$\leq$" and "$\geq$" constraints (i.e., *inequality* constraints) are associated with a half-plane of $\mathbb{R}^2$
- "$=$" constraints (or *equality* constraints) are associated with a line on the $\mathbb{R}^2$ plane.

Graphical solution of LP problems

Intersect all constraints $\Rightarrow$ the feasible set is a polyhedron.
$\Rightarrow$ Solve $\equiv$ find the point(s) with minimum $c_1x_1 + c_2x_2$.

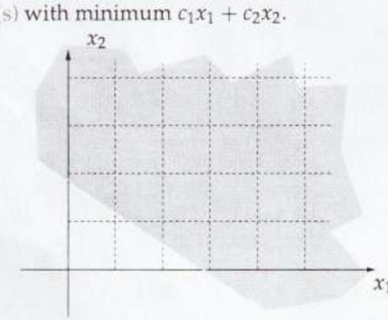
$$\begin{array}{ll}
\min & x_1 + x_2 \\
 & 2x_1 + 3x_2 \geq 6 \\
 & -x_1 + 2x_2 \leq 2 \\
 & 5x_1 + x_2 \geq 10
\end{array}$$



Graphical solution of LP problems

Intersect all constraints $\Rightarrow$ the feasible set is a polyhedron.
$\Rightarrow$ Solve $\equiv$ find the point(s) with minimum $c_1x_1 + c_2x_2$.

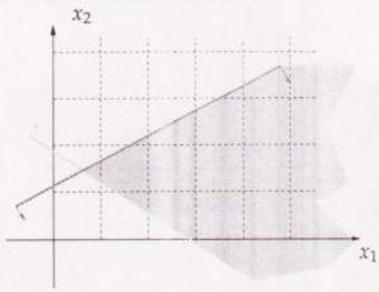
$$\begin{array}{ll}
\min & x_1 + x_2 \\
 & 2x_1 + 3x_2 \geq 6 \\
 & -x_1 + 2x_2 \leq 2 \\
 & 5x_1 + x_2 \geq 10
\end{array}$$



Graphical solution of LP problems

Intersect all constraints ⇒ the feasible set is a polyhedron.
⇒ Solve ≡ find the point(s) with minimum $c_1x_1 + c_2x_2$.

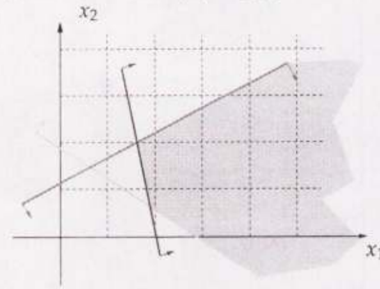
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \ge 6 \\ & -x_1 + 2x_2 \le 2 \\ & 5x_1 + x_2 \ge 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints ⇒ the feasible set is a polyhedron.
⇒ Solve ≡ find the point(s) with minimum $c_1x_1 + c_2x_2$.

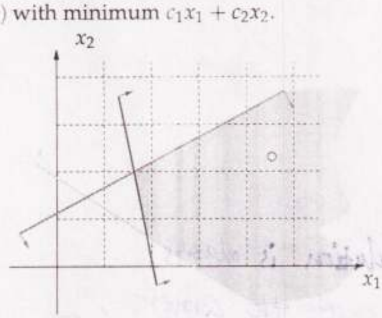
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \ge 6 \\ & -x_1 + 2x_2 \le 2 \\ & 5x_1 + x_2 \ge 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints ⇒ the feasible set is a polyhedron.
⇒ Solve ≡ find the point(s) with minimum $c_1x_1 + c_2x_2$.

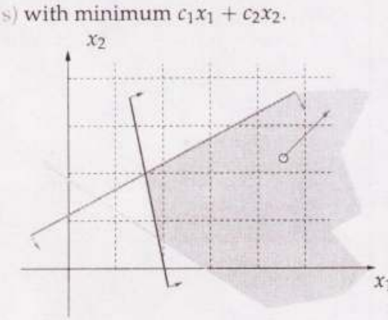
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \ge 6 \\ & -x_1 + 2x_2 \le 2 \\ & 5x_1 + x_2 \ge 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints ⇒ the feasible set is a polyhedron.
⇒ Solve ≡ find the point(s) with minimum $c_1x_1 + c_2x_2$.

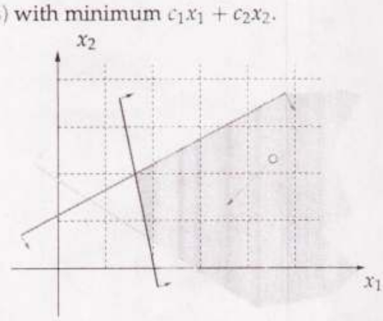
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \ge 6 \\ & -x_1 + 2x_2 \le 2 \\ & 5x_1 + x_2 \ge 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints ⇒ the feasible set is a polyhedron.
⇒ Solve ≡ find the point(s) with minimum $c_1x_1 + c_2x_2$.

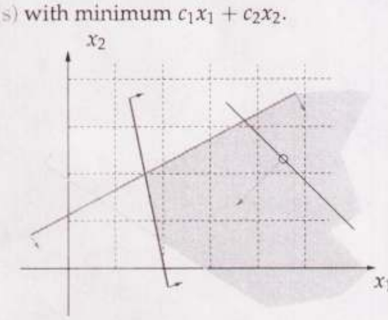
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \ge 6 \\ & -x_1 + 2x_2 \le 2 \\ & 5x_1 + x_2 \ge 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints ⇒ the feasible set is a polyhedron.
⇒ Solve ≡ find the point(s) with minimum $c_1x_1 + c_2x_2$.

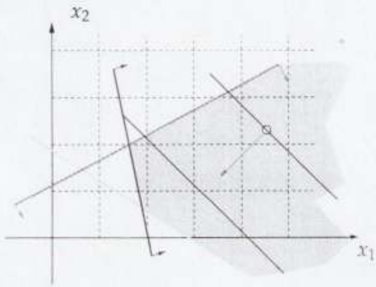
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \ge 6 \\ & -x_1 + 2x_2 \le 2 \\ & 5x_1 + x_2 \ge 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints $\Rightarrow$ the feasible set is a polyhedron.
$\Rightarrow$ Solve $\equiv$ find the point(s) with minimum $c_1x_1 + c_2x_2$.

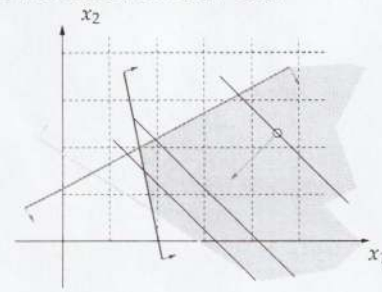
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \geq 6 \\ & -x_1 + 2x_2 \leq 2 \\ & 5x_1 + x_2 \geq 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints $\Rightarrow$ the feasible set is a polyhedron.
$\Rightarrow$ Solve $\equiv$ find the point(s) with minimum $c_1x_1 + c_2x_2$.

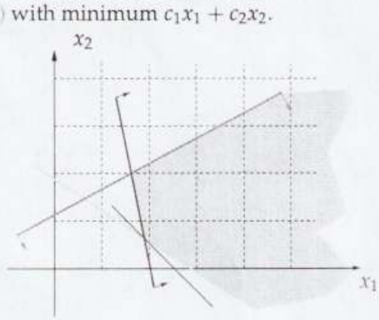
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \geq 6 \\ & -x_1 + 2x_2 \leq 2 \\ & 5x_1 + x_2 \geq 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints $\Rightarrow$ the feasible set is a polyhedron.
$\Rightarrow$ Solve $\equiv$ find the point(s) with minimum $c_1x_1 + c_2x_2$.

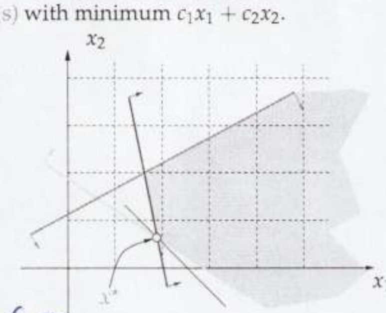
$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \geq 6 \\ & -x_1 + 2x_2 \leq 2 \\ & 5x_1 + x_2 \geq 10 \end{aligned}$$



Graphical solution of LP problems

Intersect all constraints $\Rightarrow$ the feasible set is a polyhedron.
$\Rightarrow$ Solve $\equiv$ find the point(s) with minimum $c_1x_1 + c_2x_2$.

$$\begin{aligned} \min \quad & x_1 + x_2 \\ & 2x_1 + 3x_2 \geq 6 \\ & -x_1 + 2x_2 \leq 2 \\ & 5x_1 + x_2 \geq 10 \end{aligned}$$



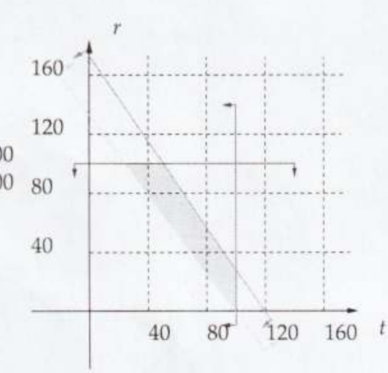
solution is always at the corner (one or two).
important

Remember the financial planning problem?

$$\begin{aligned} \max \quad & \$20t + \$15r \\ & \$50t + \$35r \leq \$6,000 \\ & \$20t + \$15r \geq \$2,000 \\ & t \leq 100 \\ & r \leq 100 \\ & t, \; r \geq 0 \end{aligned}$$

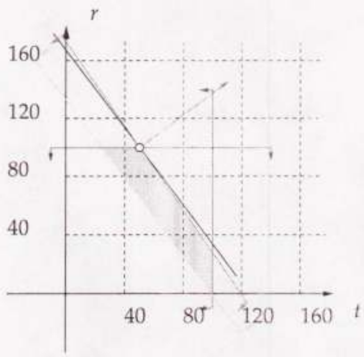
Remember the financial planning problem?

$$\begin{aligned} \max \quad & \$20t + \$15r \\ & \$50t + \$35r \leq \$6,000 \\ & \$20t + \$15r \geq \$2,000 \\ & t \leq 100 \\ & r \leq 100 \\ & t, \; r \geq 0 \end{aligned}$$



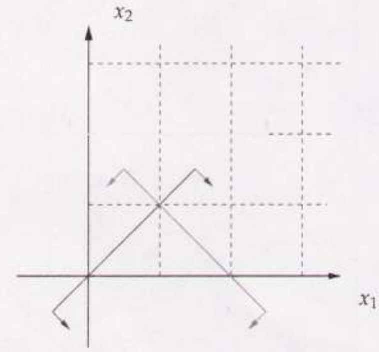
Remember the financial planning problem?

$$\begin{aligned} \max \quad & \$20t + \$15r \\ & \$50t + \$35r \le \$6,000 \\ & \$20t + \$15r \ge \$2,000 \\ & t \le 100 \\ & r \le 100 \\ & t, \; r \ge 0 \end{aligned}$$



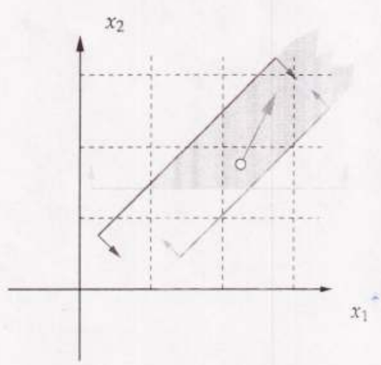
Example: Infeasible problem

$$\begin{aligned} \min \quad & x_1 + x_2 \\ & x_1 - x_2 \ge 0 \\ & x_1 + x_2 \le 2 \\ & x_2 \ge 2 \end{aligned}$$



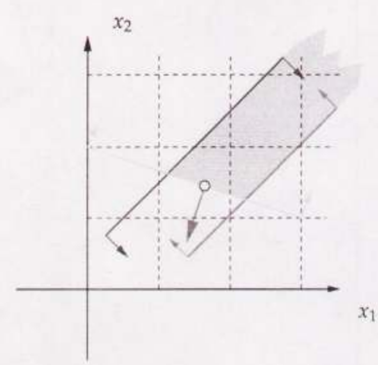
Example: Unbounded problem

$$\begin{aligned} \max \quad & x_1 + 2x_2 \\ & 2x_1 - 2x_2 \ge -1 \\ & x_1 - x_2 \le 1 \\ & x_2 \ge \tfrac{3}{2} \end{aligned}$$



Example: Multiple optima

$$\begin{aligned} \min \quad & x_1 + 3x_2 \\ & 2x_1 - 2x_2 \ge -1 \\ & x_1 - x_2 \le 1 \\ & x_1 + 3x_2 \ge 6 \end{aligned}$$



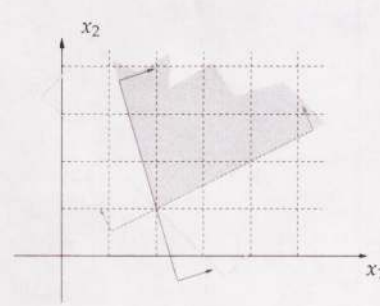
An LP problem can be...

Problems with two variables are easily classified as

- feasible and bounded (more than one optimum)
- unbounded
- infeasible

Degenerate problem

$$\begin{aligned} \min \quad & x_1 + 3x_2 \\ & x_1 - 2x_2 \le 0 \\ & 7x_1 + 2x_2 \ge 18 \\ & x_1 + x_2 \ge 3 \end{aligned}$$



ISE 426
Optimization models and applications

Lecture 5 — September 10, 2015

- Graphical solutions of LPs
- Production planning problem
- Shortest path problem
- Optimization on graphs

Reading:
WV 3.10, 7.1 and 8.2

Homework #1 will be out after class! It is due **Thursday, September 17.** Check the CourseSite.

Example: Transportation problem

- A large manufacturing company produces liquid nytrogen in five plants spread out in East Pennsylvania
- Each plant has a monthly production capacity

Plant	i	1	2	3	4	5
Capacity	p_i	120	95	150	120	140

- It has seven retailers in the same area
- Each retailer has a monthly demand to be satisfied

Retailer	j	1	2	3	4	5	6	7
Demand	d_j	55	72	80	110	85	30	78

- transportation between any plant i and any retailer j has a cost of c_{ij} dollars per volume unit of nytrogen
- c_{ij} is **constant** and depends on the distance between i and j

$\Rightarrow$ find how much nitrogen to be transported from each plant to each retailer

- ... while minimizing the total transportation cost

Transportation model

to find a corner, ≥35个 equations.

so: at least ($\cancel{35}$ 35 − 12 = 23) $x_{ij} = 0$.

Variables: qty from plant i to retailer j: x_{ij} (non-negative)

Constraints:

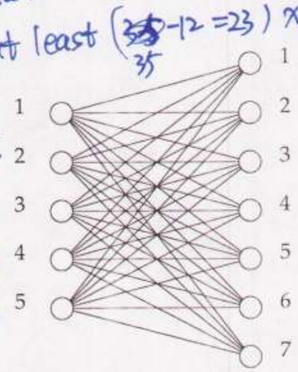
1. capacity: $\sum_{j=1}^{7} x_{ij} \le p_i \quad \forall i$
2. demand: $\sum_{i=1}^{5} x_{ij} \ge d_j \quad \forall j$

12个.

Objective function: total transportation cost, $\sum_{i=1}^{5}\sum_{j=1}^{7} c_{ij}x_{ij}$

$x_{ij} \ge 0$

35个



n x, n constraints to get 1 corner.

which means there are more than an n equations

Example: Production planning

A small firm produces plastic for the car industry.

- At the beginning of the year, it knows exactly the demand d_i of plastic for every month i.
- It also has a maximum production capacity of P and an inventory capacity of C.
- The inventory is empty on 01/01 and has to be empty again on 12/31
- production has a monthly cost c_i

What do we produce at each month to minimize total production cost while satisfying demand?

Production planning. What variables?

- How much to produce each month i: x_i, $i = 1, 2 \ldots, 12$
- Anything else?

Production planning. What variables?

- How much to produce each month i: x_i, $i = 1, 2 \ldots, 12$
- Anything else?
- The inventory level at the beginning of each month (demand is satisfied by part of production **and** part of inventory): y_i, $i = 0, 1, 2 \ldots, 12$
- Why 0? Because we need to know (actually we need to constrain!) the inventory on 01/01 and on 12/31.

Production planning. What constraints?

- Production capacity constraint: $x_i \leq P \quad \forall i = 1, 2 \ldots, 12$

Production planning. What constraints?

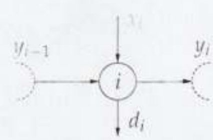
- Production capacity constraint: $x_i \leq P \quad \forall i = 1, 2 \ldots, 12$
- Inventory capacity constraint: $y_i \leq C \quad \forall i = 1, 2 \ldots, 11$

Production planning. What constraints?

- Production capacity constraint: $x_i \leq P \quad \forall i = 1, 2 \ldots, 12$
- Inventory capacity constraint: $y_i \leq C \quad \forall i = 1, 2 \ldots, 11$
- Beginning/end of year: $y_0 = 0, \quad y_{12} = 0$

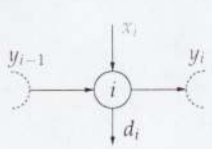
Production planning. What constraints?

- Production capacity constraint: $x_i \leq P \quad \forall i = 1, 2 \ldots, 12$
- Inventory capacity constraint: $y_i \leq C \quad \forall i = 1, 2 \ldots, 11$
- Beginning/end of year: $y_0 = 0, \quad y_{12} = 0$
- What goes in must go out...



Production planning. What constraints?

- Production capacity constraint: $x_i \leq P \quad \forall i = 1, 2 \ldots, 12$
- Inventory capacity constraint: $y_i \leq C \quad \forall i = 1, 2 \ldots, 11$
- Beginning/end of year: $y_0 = 0, \quad y_{12} = 0$
- What goes in must go out...

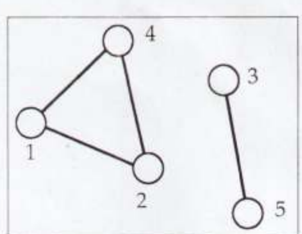


$\Rightarrow \quad x_i + y_{i-1} = d_i + y_i \quad \forall i = 1, 2 \ldots, 12$

Optimization on graphs

A (undirected) graph is defined as

- a set V of nodes (or *vertices*)
- a set E of *edges*
- each edge is a **subset** containing two nodes of V

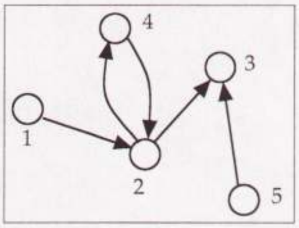


$V = \{1, 2, 3, 4, 5\}$
$E = \{\{1, 2\}, \{1, 4\}, \{2, 4\}, \{3, 5\}\}$

Directed graphs

A directed graph (or *digraph*) is defined as

- a set V of nodes (or *vertices*)
- a set A of *arcs*
- each arc is an **ordered pair** of nodes of V



$V = \{1, 2, 3, 4, 5\}$
$A = \{(1, 2), (2, 3), (4, 2), (2, 4), (5, 3)\}$

Graphs are **useful**!

...because they can model

- road networks
- gas/oil pipelines
- telecommunication networks
- electronic circuits

Optimization problems often arise in the management of network-like structures.

$\Rightarrow$ Variables, constraints, obj. f. related to nodes and edges/arcs.

The shortest path problem

Given

- a directed graph $G = (V, A)$,
- a function $c : A \to \mathbb{R}_+$, and
- two nodes s and t of V,

The shortest path problem

Given

- a directed graph $G = (V, A)$,
- a function $c : A \to \mathbb{R}_+$, and
- two nodes s and t of V,

find a subset $P = \{(s, i_1), (i_1, i_2) \dots, (i_k, t)\}$ of A forming a path from s to t whose length, $c_{si_1} + c_{i_1 i_2} + \dots + c_{i_k t}$, is minimum.

- Countless applications, e.g. GPS navigation systems.

The shortest path problem

Given

- a directed graph $G = (V, A)$,
- a function $c : A \to \mathbb{R}_+$, and
- two nodes s and t of V,

find a subset $P = \{(s, i_1), (i_1, i_2) \dots, (i_k, t)\}$ of A forming a path from s to t whose length, $c_{si_1} + c_{i_1 i_2} + \dots + c_{i_k t}$, is minimum.

- Countless applications, e.g. GPS navigation systems.

- Variables - $x_{ij} = \begin{cases} 1 & \text{if travel from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$

The shortest path problem

Given

- a directed graph $G = (V, A)$,
- a function $c : A \to \mathbb{R}_+$, and
- two nodes s and t of V,

find a subset $P = \{(s, i_1), (i_1, i_2) \dots, (i_k, t)\}$ of A forming a path from s to t whose length, $c_{si_1} + c_{i_1 i_2} + \dots + c_{i_k t}$, is minimum.

- Countless applications, e.g. GPS navigation systems.
- Variables - $x_{ij} = \begin{cases} 1 & \text{if travel from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$

$$
\begin{array}{lll}
\min & \sum_{(i,j) \in A} c_{ij} x_{ij} & \\
\text{s.t.} & \sum_{j \in V : (i,j) \in A} x_{ij} - \sum_{j \in V : (j,i) \in A} x_{ji} = b_i & \forall i \in V \\
& x_{ij} \geq 0 & \forall (i,j) \in A
\end{array}
$$

where $b_i = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}$

An example: the shortest path problem

For simplicity, the graph below is undirected, but we can assume for each edge there are two oppositely oriented arcs.

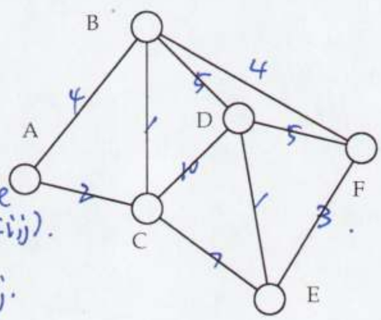
Suppose the problem is to compute the shortest path A → F.

$x_{ij} = \begin{cases} 0 \\ 1 \end{cases}$

$\forall \{i,j\} \in E$.

C_{ij} - length of edge (i,j).

$\min \sum_{(i,j) \in E} C_{ij} x_{ij}$.



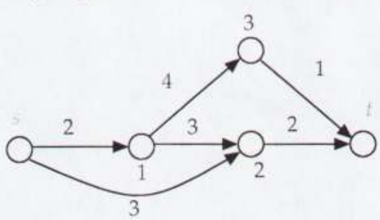
The shortest path problem: primal

$$
\begin{array}{llll}
\min & c_{AB}x_{AB} + c_{BA}x_{BA} + \ldots + c_{EF}x_{EF} & +c_{FE}x_{FE} & \\
& x_{AB} + x_{AC} & -x_{BA} - x_{CA} & = 1 \\
& x_{BA} + x_{BC} + x_{BD} + x_{BF} & -x_{AB} - x_{CB} - x_{DB} - x_{FB} & = 0 \\
& x_{CA} + x_{CB} + x_{CD} + x_{CE} & -x_{AC} - x_{BC} - x_{DC} - x_{EC} & = 0 \\
& x_{DB} + x_{DC} + x_{DE} + x_{DF} & -x_{BD} - x_{CD} - x_{ED} - x_{FD} & = 0 \\
& x_{EC} + x_{ED} + x_{EF} & -x_{CE} - x_{DE} - x_{FE} & = 0 \\
& x_{FB} + x_{FD} + x_{FE} & -x_{BF} - x_{DF} - x_{DF} & = -1 \\
& x_{AB}, x_{BA} \ldots, x_{EF}, x_{FE} \geq 0 & &
\end{array}
$$

- We can express this as $\min\{c^\top x : Ax = b, x \geq 0\}$
- A is the adjacency matrix of G
- $|V|$ constraints, $|A|$ variables
- All constraints are equalities

Problem 1: oil pipeline[2]

An oil pipeline pumps oil from an oil well s to an oil refinery t.



Each pipe has its own monthly capacity (in mega-barrels[1]). Assuming for now an infinite supply of oil at s,

⇒ maximize the amount of oil arriving at t each month

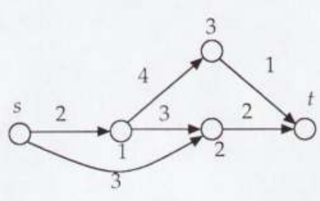
- while not exceeding pipe capacities

[1]One mega-barrel = 10^6 barrels
[2]Winston&Venkataramanan, page 420.

Max-Flow

- The Maximum Flow problem is classical in Optimization
- Can be solved with a very simple and neat algorithm
- We'll see a Linear Programming **model** of this problem
- Variables: oil flowing between each node pair
- Constraints: Oil is conserved at intermediate nodes;
  There is a maximum capacity on each pipe.
- Objective function: The total oil arriving at t
  (note: this is exactly the amount of oil that left s)

Max-Flow



- Variables: oil flowing on each arc:
  $x_{s1}, x_{s2}, x_{12}, x_{13}, x_{2t}, x_{3t}$
- Constraints: oil is conserved at intermediate nodes
  i.e. what enters node 1 exits node 1: $x_{s1} = x_{12} + x_{13}$
  what enters node 2 exits node 2: $x_{s2} + x_{12} = x_{2t}$
  what enters node 3 exits node 3: $x_{13} = x_{3t}$
- Constraints: there is a maximum capacity on each pipe,
  $x_{s1} \leq 2, x_{s2} \leq 3, x_{12} \leq 3, x_{13} \leq 4, x_{2t} \leq 2, x_{3t} \leq 1$
- Objective function: The total oil at node t: $x_{2t} + x_{3t}$
- this should be the same oil that left s: $x_{s1} + x_{s2}$

Max-Flow: the model

$$
\begin{array}{rl}
\max & x_{2t} + x_{3t} \\
x_{s1} & = x_{12} + x_{13} \\
x_{s2} + x_{12} & = x_{2t} \\
x_{13} & = x_{3t} \\
0 \leq x_{s1} & \leq 2 \\
0 \leq x_{s2} & \leq 3 \\
0 \leq x_{12} & \leq 3 \\
0 \leq x_{13} & \leq 4 \\
0 \leq x_{2t} & \leq 2 \\
0 \leq x_{3t} & \leq 1
\end{array}
$$

不能相加消元. 否则 constraints 消失.

Could re-write objective function as $\max\ x_{s1} + x_{s2}$

Max-Flow: the general model

Consider a digraph $G = (V, A)$:

- two nodes s and t of V act as *source* and *destination*.
- each arc $(i,j) \in A$ has a capacity c_{ij}

Max-Flow: the general model

Consider a digraph $G = (V, A)$:

- two nodes s and t of V act as *source* and *destination*.
- each arc $(i,j) \in A$ has a capacity c_{ij}
- Variables: flow on each arc (i,j), call it x_{ij}
  In general, x_{ij} can be $\neq x_{ji}$ (opposite flow)

Max-Flow: the general model

Consider a digraph $G = (V, A)$:

- two nodes s and t of V act as *source* and *destination*.
- each arc $(i,j) \in A$ has a capacity c_{ij}
- Variables: flow on each arc (i,j), call it x_{ij}
  In general, x_{ij} can be $\neq x_{ji}$ (opposite flow)
- Constraints: conservation of flow at intermediate node i.

Max-Flow: the general model

Consider a digraph $G = (V, A)$:

- two nodes s and t of V act as *source* and *destination*.
- each arc $(i,j) \in A$ has a capacity c_{ij}
- Variables: flow on each arc (i,j), call it x_{ij}
  In general, x_{ij} can be $\neq x_{ji}$ (opposite flow)
- Constraints: conservation of flow at intermediate node i.

At an interm. node, $\forall i \in V : s \neq i \neq t$ — what enters — will leave

Max-Flow: the general model

Consider a digraph $G = (V, A)$:

- two nodes s and t of V act as *source* and *destination*.
- each arc $(i,j) \in A$ has a capacity c_{ij}
- Variables: flow on each arc (i,j), call it x_{ij}
  In general, x_{ij} can be $\neq x_{ji}$ (opposite flow)
- Constraints: conservation of flow at intermediate node i.

At an interm. node,	what enters		will leave
$\forall i \in V : s \neq i \neq t$	$\sum_{j \in V:(j,i) \in A} x_{ji}$	$=$	$\sum_{j \in V:(i,j) \in A} x_{ij}$

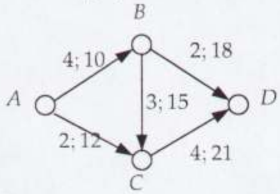
- Constraints: min and max flow, $0 \leq x_{ij} \leq c_{ij} \forall (i,j) \in A$

Max-Flow: the general model

- Objective Function: flow entering t, i.e., $\sum_{j \in V:(j,t) \in A} x_{jt}$
- (same that leaves s, i.e. $\sum_{j \in V:(s,j) \in A} x_{sj}$)

$$\begin{array}{lll} \max & \sum_{j \in V:(j,t) \in A} x_{jt} & \\ \text{s.t.} & \sum_{j \in V:(j,i) \in A} x_{ji} = \sum_{j \in V:(i,j) \in A} x_{ij} & \forall i \in V : s \neq i \neq t \\ & 0 \leq x_{ij} \leq c_{ij} & \forall (i,j) \in A \end{array}$$

Problem 2: another oil pipeline



- Oil company X now uses company Y's pipeline
- Each month, X wants to pump 5 mega-barrels from A to D,
- Y reserves part of the capacity and charges a cost/flow

A→B allows ≤ 4 mega-barrels, cost of 10k$/mega-barrel
A→C allows ≤ 2 at a cost of 12
B→C allows ≤ 3, cost: 15
B→D allows ≤ 2, cost: 18
C→D allows ≤ 4, cost: 21

⇒ Can the company send oil from A to D?

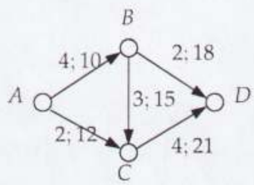
- How to do it at minimum cost?

Minimum cost flow

Another classical problem in Optimization

- Variables: same as in the Max-Flow problem, i.e., quantity of oil flowing on each arc (i.e. between each node pair)
- Constraints: same as in the Max-Flow problem, plus:
- the flow leaving node A (= entering node D) has to be equal to 5 mega-barrels
- Objective function: total flow cost

Min-Cost-Flow



- Variables: oil flowing on each arc:
  $x_{AB}, x_{AC}, x_{BC}, x_{BD}, x_{CD}$
- Constraints: oil does not evaporate at interm. nodes
  i.e. what enters B exits B: $x_{AB} = x_{BC} + x_{BD}$
  what enters C exits C: $x_{AC} + x_{BC} = x_{CD}$
- Constraints: there is a maximum capacity on each pipe,
  $x_{AB} \leq 4$, $x_{AC} \leq 2$, $x_{BC} \leq 3$, $x_{BD} \leq 2$, $x_{CD} \leq 4$
- Constraint: required flow must leave A, i.e., $x_{AB} + x_{AC} = 5$
- Objective function: The total pumping cost:
  $10x_{AB} + 12x_{AC} + 15x_{BC} + 18x_{BD} + 21x_{CD}$

Min-Cost-Flow: the model

$$\min \quad 10x_{AB} + 12x_{AC} + 15x_{BC} + 18x_{BD} + 21x_{CD}$$

$$x_{AB} = x_{BC} + x_{BD}$$
$$x_{AC} + x_{BC} = x_{CD}$$
$$x_{AB} + x_{AC} = 5$$

$$0 \leq x_{AB} \leq 4$$
$$0 \leq x_{AC} \leq 2$$
$$0 \leq x_{BC} \leq 3$$
$$0 \leq x_{BD} \leq 2$$
$$0 \leq x_{CD} \leq 4$$

Min-Cost-Flow: the general model

Consider a digraph $G = (V, A)$:

- two nodes s and t of V act as *source* and *destination*.
- each arc $(i, j) \in A$ has a capacity c_{ij} and a cost d_{ij}
- a *required flow* r
- Variables: flow on each arc (i, j), call it x_{ij}
- Constraints: conservation of flow at intermediate node i:

At an interm. node,	what enters		must leave
$\forall i \in V : s \neq i \neq t$	$\sum_{j \in V:(j,i) \in A} x_{ji}$	$=$	$\sum_{j \in V:(i,j) \in A} x_{ij}$

- Constraints: min and max flow, $0 \leq x_{ij} \leq c_{ij}$ $\forall (i, j) \in A$
- Constraint: required flow leaves s, i.e., $\sum_{j \in V:(s,j) \in A} x_{sj} = r$

Min-Cost-Flow: the general model

- Objective Function: cost of flow, i.e., $\sum_{(i,j) \in A} d_{ij} x_{ij}$

$$\begin{array}{lll} \min & \sum_{(i,j) \in A} d_{ij} x_{ij} & \\ \text{s.t.} & \sum_{j \in V:(j,i) \in A} x_{ji} = \sum_{j \in V:(i,j) \in A} x_{ij} & \forall i \in V : s \neq i \neq t \\ & \sum_{j \in V:(s,j) \in A} x_{sj} = r & \\ & 0 \leq x_{ij} \leq c_{ij} & \forall (i,j) \in A \end{array}$$

Min-Cost-Flow: the general model with multiple sources and sinks

Consider a digraph $G = (V, A)$:

- Sets of nodes $s \in S \subseteq V$ and $t \in T \subseteq V$ act as *sources* and *destinations*.
- each arc $(i, j) \in A$ has a capacity c_{ij} and a cost d_{ij}
- Each destination $t \in T$ has a *required flow* demand d_t
- Variables: flow on each arc (i, j), call it x_{ij}
- Constraints: conservation of flow at intermediate node i:

At an interm. node,	what enters		must leave
$\forall i \in V/\{S \cup T\}$	$\sum_{j \in V:(j,i) \in A} x_{ji}$	$=$	$\sum_{j \in V:(i,j) \in A} x_{ij}$

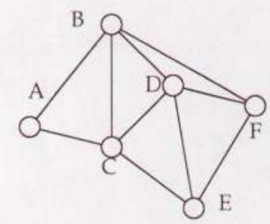
- Constraints: min and max flow, $0 \leq x_{ij} \leq c_{ij}$ $\forall (i, j) \in A$
- Constraints: required flow demand for each $t \in T$, i.e., $\sum_{j \in V:(j,t) \in A} x_{jt} - \sum_{j \in V:(t,j) \in A} x_{tj} = d_t$

What about the shortest path problem?

$$\begin{array}{lll} \min & \sum_{(i,j) \in A} c_{ij} x_{ij} & \\ \text{s.t.} & \sum_{j \in V:(i,j) \in A} x_{ij} - \sum_{j \in V:(j,i) \in A} x_{ji} = b_i & \forall i \in V \\ & x_{ij} \geq 0 & \forall (i,j) \in A \end{array}$$

$$\text{where } b_i = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}$$

The problem is to compute the shortest path A → F.



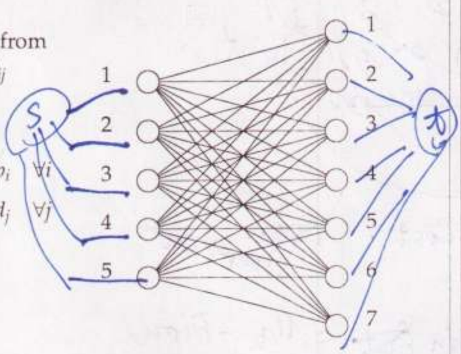
Transportation model?

Variables: qty of product from producer i to location j: x_{ij} (non-negative)

Constraints:

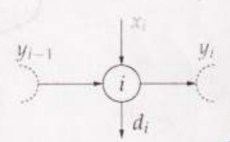
1. capacity: $\sum_{j=1}^{7} x_{ij} \leq p_i$ $\forall i$
2. demand: $\sum_{i=1}^{5} x_{ij} \geq d_j$ $\forall j$

Objective function: total transportation cost, $\sum_{i=1}^{5} \sum_{j=1}^{7} c_{ij} x_{ij}$

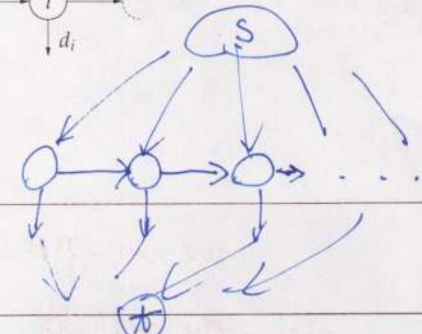


Production planning?

- Production capacity constraint: $x_i \leq P$ $\forall i = 1, 2 \ldots, 12$
- Beginning/end of year: $y_0 = 0$, $y_{12} = 0$
- What goes in must go out…



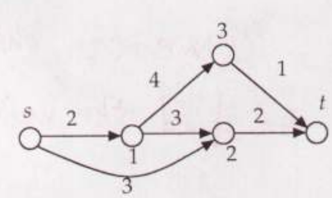
$\Rightarrow$ $x_i + y_{i-1} = d_i + y_i$ $\forall i = 1, 2 \ldots, 12$



Generalizations

- multiple sources (have a negative flow balance), multiple destinations (positive flow balance)
- non-zero lower bounds on flows
- nonlinear flow costs
- multiple types of flow on the same arc

Back to Max-Flow



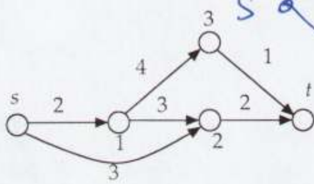
- Objective Function: flow entering t, i.e., $\sum_{j \in V:(j,t) \in A} x_{jt}$
- (same that leaves s, i.e. $\sum_{j \in V:(s,j) \in A} x_{sj}$)

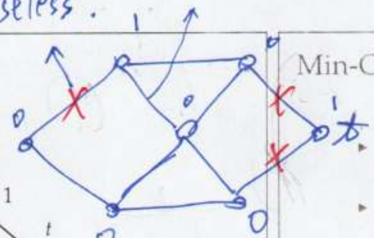
$$\begin{array}{lll} \max & \sum_{j \in V:(j,t) \in A} x_{jt} & \\ \text{s.t.} & \sum_{j \in V:(j,i) \in A} x_{ji} = \sum_{j \in V:(i,j) \in A} x_{ij} & \forall i \in V : s \neq i \neq t \\ & 0 \leq x_{ij} \leq c_{ij} & \forall (i,j) \in A \end{array}$$

1 → 0 no cut.

useless.

Min-Cut





- What is the set of edges of the smallest capacity which, if cut, completely cuts off s from t?
- Useful applications: finding bottleneck in networks, damming systems of rivers, military...

in some feasible solution

there ~~are man~~ may be some useless cut.

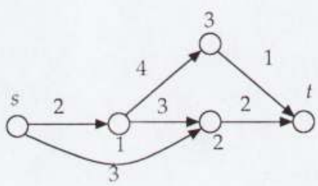
Min-Cut

- Variables for each node $u_i = \{0,1\}$: 0 if i is cut off from t, 1 if i is not cut off from t.
- Variables for each arc $z_{ij} = \{0,1\}$: 1 if arc (i,j) is in the cut, 0 otherwise.
- Objective Function: the capacity of the cut $\sum_{(i,j)\in A} c_{ij}z_{ij}$

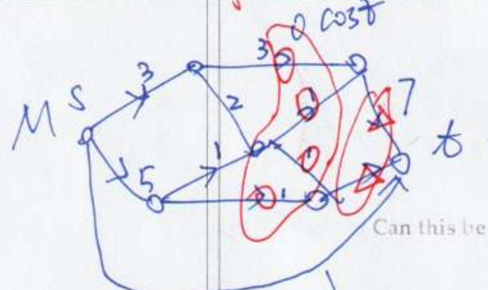
$$
\begin{array}{lll}
\text{min.}\ \cancel{\max} & \sum_{(i,j)\in A} c_{ij}z_{ij} & \\
\text{s.t.} & z_{ij} \geq u_j - u_i & \\
 & u_s = 0 & \\
 & u_t = 1 & \\
 & 0 \leq z_{ij} & \forall (i,j) \in A \\
 & 0 \leq u_i & \forall i \in V
\end{array}
$$

just like cut sets in 419.

Min-Cut = Max-Flow



- **THEOREM:** (Ford, Fulkerson, 1956) The capacity of the minimum cut equals the value of the maximum flow
- **NOTE:** The capacity of any cut is gearter or equal to the value of any feasible flow in the network.



Can this be done for general LP problems?

a Cost. ~~Cut~~ shortcut.

Min ~~Cut~~ Shortcut = Max-Flow

we can ~~take~~ take each flow as a job, capacity as minus Time.

then, the graph is "job on arc".

we can change it to "job on node".

to find the limited l_{ij}, find the cut sets firstly.

ISE 426
Optimization models and applications

Lecture 8 — September 22, 2015

Duality

Reading:

- W.&V. Sections 6.5–6.7, pages 295-308
- H.&L. Section 6.1–6.4, pages 151-169

First: playing with equations and inequalities

- Trivial: if $a \leq b$ and $c \leq d$, then $\boxed{a + c \leq b + d}$
- Similarly, if $a = b$ and $c = d$, then $\boxed{a + c = b + d}$
- And: if $a \leq b$ and $c = d$, then $\boxed{a + c \leq b + d}$
- Also, if $a \leq b$, for any $k \geq 0$ we have $\boxed{ka \leq kb}$

$\Rightarrow$ We can mix inequalities (and equations)! Example:

$$\begin{aligned} a_1x_1 + a_2x_2 \ldots + a_nx_n &\leq a_0 \\ b_1x_1 + b_2x_2 \ldots + b_nx_n &\leq b_0 \\ c_1x_1 + c_2x_2 \ldots + c_nx_n &\leq c_0 \end{aligned}$$

and three numbers, $p, q, r \geq 0$, the following is true:

$$\begin{aligned} &p(a_1x_1 + a_2x_2 \ldots + a_nx_n) + \\ &q(b_1x_1 + b_2x_2 \ldots + b_nx_n) + \\ &r(c_1x_1 + c_2x_2 \ldots + c_nx_n) \qquad \leq pa_0 + qb_0 + rc_0 \end{aligned}$$

That is, a nonnegative combination of constraints is valid.

Second: non-negative variables and linear functions

Consider $f(x_1, x_2) = 5x_1 + 3x_2$, and suppose $x_1, x_2 \geq 0$.
What is $\leq f(x_1, x_2)$ for any $x_1, x_2 \geq 0$?

- $g_1(x_1, x_2) = 0$
- $g_2(x_1, x_2) = x_1$
- $g_3(x_1, x_2) = 5x_1$
- $g_4(x_1, x_2) = 5.00001x_1$
- $g_5(x_1, x_2) = 4x_1 + 2x_2$
- $g_6(x_1, x_2) = 2x_1 + 9x_2$
- $g_7(x_1, x_2) = 3x_1 + 5x_2$

What can we conclude?

Second: non-negative variables and linear functions

Consider $f(x_1, x_2) = 5x_1 + 3x_2$, and suppose $x_1, x_2 \geq 0$.
What is $\leq f(x_1, x_2)$ for any $x_1, x_2 \geq 0$?

- $g_1(x_1, x_2) = 0$
- $g_2(x_1, x_2) = x_1$
- $g_3(x_1, x_2) = 5x_1$
- $g_4(x_1, x_2) = 5.00001x_1$
- $g_5(x_1, x_2) = 4x_1 + 2x_2$
- $g_6(x_1, x_2) = 2x_1 + 9x_2$
- $g_7(x_1, x_2) = 3x_1 + 5x_2$

What can we conclude?
For $x_1, x_2 \geq 0$, a function $g(x_1, x_2) = ax_1 + bx_2$ is lower than $f(x_1, x_2)$ only if $a \leq 5$ and $b \leq 3$.

Lower bounds of an LP problem

Consider the following **minimization** problem:

$$\begin{array}{ll} \min & 5x_1 + 4x_2 \\ & 2x_1 + x_2 \quad \geq 1 \\ & x_1 + 2x_2 \quad \geq 1 \\ & x_1, x_2 \geq 0 \end{array}$$

- Solve this problem graphically
- Using constraints can we derive lower bounds for the optimal value?

Lower bounds of an LP problem

$$\begin{array}{ll} \min & 5x_1 + 4x_2 \\ & 2x_1 + x_2 \quad \geq 1 \\ & x_1 + 2x_2 \quad \geq 1 \\ & x_1, x_2 \geq 0 \end{array}$$

- The optimal solution $x^* = (1/3, 1/3)$, $f(x^*) = 5x_1^* + 4x_2^* = 3$.
- $2x_1 + x_2 \geq 1$, and $x_1, x_2 \geq 0$ hence $5x_1^* + 4x_2^* \geq 1$.
- $2x_1 + x_2 \geq 1$, and $x_1 + 2x_2 \geq 1$ hence $5x_1^* + 4x_2^* \geq 2$.
- $2(2x_1 + x_2) \geq 2$, and $x_1 + 2x_2 \geq 1$ hence $5x_1^* + 4x_2^* \geq 3$.
- Once we know that $5x_1^* + 4x_2^* \geq 3$ and that $x = (1/3, 1/3)$ is feasible, then we know it is optimal.

Lower bounds of an LP problem, another example

Consider the following **minimization** problem:

$$\begin{array}{ll} \min & 3x_1 + 4x_2 \\ & 5x_1 + 6x_2 \geq 7 \\ & 8x_1 + 9x_2 \geq 10 \\ & 11x_1 + 12x_2 \geq 13 \\ & x_1, x_2 \geq 0 \end{array}$$

We have a lower bound if we prove $3x_1 + 4x_2 \geq K$, for some K, by showing that

$$ax_1 + bx_2 \geq K$$

with $a \leq 3$ and $b \leq 4$.

Finding lower bounds using constraints

We can use nonnegative combinations of constraints?

For example: 0.25(first constraint) + 0.2(second constraint) =

$$\begin{array}{llll} 0.25(5x_1+6x_2) & +0.2(8x_1+9x_2) & \geq & 0.25\cdot 7 + 0.2\cdot 10 \\ (1.25+1.6)x_1 & +(1.5+1.8)x_2 & \geq & 1.75+2.0 \\ 2.85x_1 & +3.3x_2 & \geq & 3.75 \end{array}$$

- $2.85 \leq 3$ and $3.3 \leq 4$

$\Rightarrow$ $2.85x_1 + 3.3x_2$ is a lower bound of $3x_1 + 4x_2$ for $x_1, x_2 \geq 0$

$\Rightarrow$ 3.75 is a lower bound:

$$3x_1 + 4x_2 \geq 2.85x_1 + 3.3x_2 \geq 3.75$$

So what kind of inequalities can we get?

- We have three constraints to work with
- We need three numbers $u_1, u_2, u_3 \geq 0$
- With u_1, u_2, u_3, we construct a new constraint $ax_1 + bx_2 \geq c$:

$$u_1(5x_1+6x_2) + u_2(8x_1+9x_2) + u_3(11x_1+12x_2) \geq 7u_1 + 10u_2 + 13u_3$$
$$\underbrace{(5u_1+8u_2+11u_3)}_{a}x_1 + \underbrace{(6u_1+9u_2+12u_3)}_{b}x_2 \geq \underbrace{7u_1+10u_2+13u_3}_{c}$$

- The new constraint is valid for any $u_1, u_2, u_3 \geq 0$
- We want $ax_1 + bx_2$ to be always below $3x_1 + 4x_2$

... (so that $c = 7u_1 + 10u_2 + 13u_3$ is a valid lower bound)

$\Rightarrow$ We must have $a \leq 3$ and $b \leq 4$. Hence,

$$\begin{array}{l} 5u_1 + 8u_2 + 11u_3 \leq 3 \\ 6u_1 + 9u_2 + 12u_3 \leq 4 \end{array}$$

To recap

For any u_1, u_2, u_3 such that

$$\begin{array}{l} 5u_1 + 8u_2 + 11u_3 \leq 3 \\ 6u_1 + 9u_2 + 12u_3 \leq 4 \\ u_1, u_2, u_3 \geq 0 \end{array}$$

$7u_1 + 10u_2 + 13u_3$ is a lower bound of $3x_1 + 4x_2$. Examples:

- $(u_1, u_2, u_3) = (0, 0, 0)$. Lower bound: 0
- $(u_1, u_2, u_3) = (0.2, 0.1, 0.1)$. Lower bound: 3.7
- $(u_1, u_2, u_3) = (0.1, 0.2, 0)$. Lower bound: 2.7
- $(u_1, u_2, u_3) = (0.5, 0, 0.04)$. Lower bound: 4.

Should we try all possible combinations? Is there a better way?

The dual problem

Yes, there is. We like large lower bounds, so we want to maximize that $7u_1 + 10u_2 + 13u_3$.

$$\begin{array}{ll} \max & 7u_1 + 10u_2 + 13u_3 \\ & 5u_1 + 8u_2 + 11u_3 \leq 3 \\ & 6u_1 + 9u_2 + 12u_3 \leq 4 \\ & u_1, u_2, u_3 \geq 0 \end{array}$$

- Any feasible solution to this problem provides a lower bound to the original problem.
- An optimal solution to this problem provides a **good** lower bound to the original problem.

Primal problem, dual problem

Primal	Dual
$\begin{array}{ll} \min & 3x_1 + 4x_2 \\ & 5x_1 + 6x_2 \geq 7 \\ & 8x_1 + 9x_2 \geq 10 \\ & 11x_1 + 12x_2 \geq 13 \\ & x_1, x_2 \geq 0 \end{array}$	$\begin{array}{ll} \max & 7u_1 + 10u_2 + 13u_3 \\ & 5u_1 + 8u_2 + 11u_3 \leq 3 \\ & 6u_1 + 9u_2 + 12u_3 \leq 4 \\ & u_1, u_2, u_3 \geq 0 \end{array}$

In general:

$\begin{array}{ll} \min & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{array}$	$\begin{array}{ll} \max & b^\top u \\ & A^\top u \leq c \\ & u \geq 0 \end{array}$

The **primal** has n variables and m constraints

$\Rightarrow$ The **dual** has m variables and n constraints

Primal problem, dual problem

Primal		Dual	
min $5x_1 + 4x_2$		max $u_1 + u_2$	
$2x_1 + x_2$	≥ 1	$2u_1 + u_2$	≤ 5
$x_1 + 2x_2$	≥ 1	$u_1 + 2u_2$	≤ 4
$x_1, x_2 \geq 0$		$u_1, u_2 \geq 0$	

is important ~~prerequisit~~ premise

$\min\ 5x_1 + 4x_2 \qquad -2x_1 - x_2 \geq -1$

$2x_1 + x_2 \leq 1 \quad u_1 \geq 0 \rightarrow \begin{cases} 2x_1 + x_2 \leq 1 & u_1 \leq 0 \\ x_1 + 2x_2 \geq 1 & u_2 \geq 0 \end{cases}$ $\boxed{\text{bad}}$

$x_1 + 2x_2 \geq 1 \quad u_2 \geq 0$

$x_1, x_2 \geq 0.$

max $-u_1 + u_2$

$-2u_1 + u_2 \leq 5, \ x_1 \geq 0$

$-u_1 + 2u_2 \leq 4, \ x_2 \geq 0$

$u_1, u_2 \geq 0$

then, let $\bar{u}_1 = -u_1 \leq 0.$

max $\bar{u}_1 + u_2$

$2\bar{u}_1 + u_2 \leq 5$

$\bar{u}_1 + 2u_2 \leq 4$

How to construct the dual of an LP

Variable	Constraint
Constraint	Variable
Minimize	Maximize
Variable ≥ 0	Constraint $\leq$
Variable ≤ 0	Constraint $\geq$
Var. Unrestricted	Constraint $=$
Constraint $\leq$	Variable ≤ 0
Constraint $\geq$	Variable ≥ 0
Constraint $=$	Var. Unrestricted

Examples

$$\begin{array}{llll} \min & 12x_1 & -47x_2 & \\ \text{s.t.} & 25x_1 & -36x_2 & \leq 97 \\ & 38x_1 & +89x_2 & \geq 10 \\ & x_1 \geq 0 & & \end{array} \rightarrow \begin{array}{llll} \max & 97u_1 & +10u_2 & \\ \text{s.t.} & 25u_1 & +38u_2 & \leq 12 \\ & -36u_1 & +89u_2 & = -47 \\ & u_1 \leq 0 & u_2 \geq 0 & \end{array}$$

$$\begin{array}{ll} \max & 2x_1 + x_2 \\ \text{s.t.} & -x_1 + x_2 = 9 \\ & 9x_1 - x_2 \geq 11 \\ & 7x_1 + 2x_2 \leq 3 \\ & -x_1 + 7x_2 \geq 1 \\ & x_1 \leq 0, x_2 \geq 0 \end{array} \rightarrow \begin{array}{ll} \min & 9u_1 + 11u_2 + 3u_3 + u_4 \\ \text{s.t.} & -u_1 + 9u_2 + 7u_3 - u_4 \leq 2 \\ & u_1 - u_2 + 2u_3 + 7u_4 \geq 1 \\ & u_2, u_4 \leq 0; u_3 \geq 0 \end{array}$$

What is the dual of the dual?

if $\begin{cases} 2x_1 + x_2 = 1, & u_1 \\ x_1 + 2x_2 \geq 1, & u_2 \geq 0 \end{cases}$

$$\begin{array}{ll} \min & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{array} \rightarrow \begin{array}{ll} \max & b^\top u \\ & A^\top u \leq c \\ & u \geq 0 \end{array} \rightarrow \begin{array}{ll} \min & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{array}$$

- The dual of the dual is the primal problem
- An LP and its dual are said to form a **primal-dual pair**

Properties of duality in LP

Weak duality: Given a primal $\min\{c^\top x : Ax \geq b, x \geq 0\}$ and its dual $\max\{b^\top u : A^\top u \leq c, u \geq 0\}$,

$$b^\top \bar{u} \leq c^\top \bar{x}$$

for any $\bar{x}$ and $\bar{u}$ feasible for their respective problems.

- Proof: $c^\top \bar{x} \geq (A^\top \bar{u})^\top \bar{x} = \bar{u}^\top A \bar{x} \geq \bar{u}^\top b$
- $u^\top b$ is a LB, our main purpose when constructing the dual.

Strong duality[1]: If a problem $\min\{c^\top x : Ax \geq b, x \geq 0\}$ is bounded and its dual $\max\{b^\top u : A^\top u \leq c, u \geq 0\}$ is bounded, their optimal solutions $\bar{x}$ and $\bar{u}$ coincide in value:

$$c^\top \bar{x} = b^\top \bar{u}$$

$\cancel{(c - A^\top \bar{u})} = 0$

$(A\bar{x} - b)\bar{u} = 0.$

[1]The proof of this is much more complicated, but beautiful nonetheless.

$\begin{cases} (c^\top - A^\top \bar{u})\bar{x} = 0 \\ \bar{u}(A\bar{x} - b) = 0 \end{cases}$

Properties of duality in LP (cont.)

Consequence: solving the dual or the primal doesn't matter: we get the same objective function value.

What if the primal (or the dual) is infeasible or unbounded?

Four cases:

- Primal bounded, dual bounded;
- Primal infeasible, dual infeasible;
- Primal unbounded ($c^\top x = -\infty$), dual infeasible;
- Primal infeasible, dual unbounded ($b^\top u = +\infty$).

		Dual		
		bounded	unbounded	infeasible
Primal	bounded	Possible	–	–
	unbounded	–	–	Possible
	infeasible	–	Possible	Possible

Complementary slackness

- Given a primal-dual pair, now we know how to solve one and get the optimal objective function of the other.

e.g. Solve primal $\Rightarrow$ get optimal obj.f. $c^\top \bar{x}$, an optimal solution $\bar{x}$, and the optimal dual obj.f. $b^\top \bar{u}$. **How do we get $\bar{u}$?**

> Complementary Slackness: If the primal problem
> $$\min\{c^\top x : \textstyle\sum_{i=1}^n a_{ji}x_i \geq b_j \ \forall j = 1,2\dots,m, x \geq 0\}$$
> is bounded and admits optimum $\bar{x}$, and its dual
> $$\max\{b^\top u : \textstyle\sum_{j=1}^m a_{ji}u_j \leq c_i \ \forall i = 1,2\dots,n, u \geq 0\}$$
> is bounded and admits optimal solution $\bar{u}$, then
> $$\bar{u}_i(\textstyle\sum_{i=1}^n a_{ji}\bar{x}_i - b_j) = 0 \quad \forall i = 1,2\dots,m;$$
> $$\bar{x}_j(\textstyle\sum_{j=1}^m a_{ji}\bar{u}_j - c_j) = 0 \quad \forall j = 1,2\dots,n$$

So if we solve the primal and get $\bar{x}$, we can get $\bar{u}$ by solving a system of equations.

Example

$$\begin{array}{llll} \min & 3x_1 + 4x_2 & \\ & 5x_1 + 6x_2 & \geq 7 \\ & 8x_1 + 9x_2 & \geq 10 \\ & 11x_1 + 12x_2 & \geq 13 \\ & x_1, x_2 \geq 0 & \end{array} \qquad \begin{array}{lll} \max & 7u_1 + 10u_2 + 13u_3 & \\ & 5u_1 + 8u_2 + 11u_3 & \leq 3 \\ & 6u_1 + 9u_2 + 12u_3 & \leq 4 \\ & u_1, u_2, u_3 \geq 0 & \end{array}$$

Solve the dual (with AMPL+CPLEX): get $(u_1, u_2, u_3) = (0.6, 0, 0)$.
Find (x_1, x_2) with complementary slackness:

$$\begin{array}{lcl} u_1(5x_1 + 6x_2 - 7) = 0 & & 0.6(5x_1 + 6x_2 - 7) = 0 \\ u_2(8x_1 + 9x_2 - 10) = 0 & & 0(8x_1 + 9x_2 - 10) = 0 \\ u_3(11x_1 + 12x_2 - 13) = 0 & \Rightarrow & 0(11x_1 + 12x_2 - 13) = 0 \\ x_1(5u_1 + 8u_2 + 11u_3 - 3) = 0 & & x_1(5 \cdot 0.6 + 8 \cdot 0 + 11 \cdot 0 - 3) = 0 \\ x_2(6u_1 + 9u_2 + 12u_3 - 4) = 0 & & x_2(6 \cdot 0.6 + 9 \cdot 0 + 12 \cdot 0 - 4) = 0 \end{array}$$

$$\begin{array}{l} 5x_1 + 6x_2 = 7 \\ x_1 \cdot 0 = 0 \\ x_2 \cdot (-0.4) = 0 \end{array} \Rightarrow \begin{array}{l} 5x_1 + 6x_2 = 7 \\ x_1 \cdot 0 = 0 \\ x_2 = 0 \end{array} \Rightarrow \begin{array}{l} x_1 = \frac{7}{5} \\ x_2 = 0 \end{array}$$

An example: the shortest path problem

Given

- a digraph $G = (V, A)$,
- a function $c : A \to \mathbb{R}_+$, and
- two nodes s and t of V,

find a subset $P = \{(s, i_1), (i_1, i_2) \dots, (i_k, t)\}$ of A forming a path from s to t whose length, $c_{si_1} + c_{i_1 i_2} \dots c_{i_k t}$, is minimum.

- Countless applications, e.g. GPS navigation systems.
- We can formulate it as a special case of min-cost-flow:

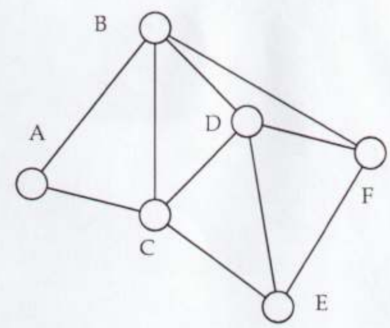
$$\begin{array}{lll} \min & \sum_{(i,j)\in A} c_{ij}x_{ij} & \\ \text{s.t.} & \sum_{j\in V:(i,j)\in A} x_{ij} - \sum_{j\in V:(j,i)\in A} x_{ji} = b_i & \forall i \in V \\ & x_{ij} \geq 0 & \forall (i,j) \in A \end{array}$$

$$\text{where } b_i = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases}$$

An example: the shortest path problem

For simplicity, the graph below is undirected, but we can assume for each edge there are two oppositely oriented arcs.

Suppose the problem is to compute the shortest path A $\to$ F.



The shortest path problem: primal

$$\begin{array}{llr} \min & c_{AB}x_{AB} + c_{BA}x_{BA} + \ldots + c_{EF}x_{EF} \quad +c_{FE}x_{FE} & \\ & x_{AB} + x_{AC} \quad -x_{BA} - x_{CA} & = 1 \\ & x_{BA} + x_{BC} + x_{BD} + x_{BF} \quad -x_{AB} - x_{CB} - x_{DB} - x_{FB} & = 0 \\ & x_{CA} + x_{CB} + x_{CD} + x_{CE} \quad -x_{AC} - x_{BC} - x_{DC} - x_{EC} & = 0 \\ & x_{DB} + x_{DC} + x_{DE} + x_{DF} \quad -x_{BD} - x_{CD} - x_{ED} - x_{FD} & = 0 \\ & x_{EC} + x_{ED} + x_{EF} \quad -x_{CE} - x_{DE} - x_{FE} & = 0 \\ & x_{FB} + x_{FD} + x_{FE} \quad -x_{BF} - x_{DF} - x_{DF} & = -1 \\ & x_{AB}, x_{BA} \dots, x_{EF}, x_{FE} \geq 0 & \end{array}$$

- We can express this as $\min\{c^\top x : Ax = b, x \geq 0\}$
- A is the adjacency matrix of G
- $|V|$ constraints, $|A|$ variables
- All constraints are equalities

The shortest path problem: dual

$$\begin{array}{ll} \max & u_A - u_F \\ & u_A - u_B \leq c_{AB} \\ & u_B - u_A \leq c_{BA} \\ & u_A - u_C \leq c_{AC} \\ & u_C - u_A \leq c_{CA} \\ & \vdots \\ & u_E - u_F \leq c_{EF} \\ & u_F - u_E \leq c_{FE} \end{array}$$

- This is $\max\{b^\top u : A^\top u \leq c\}$
- $A^\top$ is the **transposed** adjacency matrix of G
- $|V|$ variables, $|A|$ constraints
- All variables are unrestricted in sign

ISE 426
Optimization models and applications

Lecture 9 — September 29, 2015

Duality, continued

Reading:

- W.&V. Sections 6.5–6.7, pages 295-308
- H.&L. Section 6.1–6.4, pages 151-169

Reminders:

- Quiz on 10/08, practice on 10/06.

Primal problem, dual problem

Primal

$$\begin{array}{ll} \min & 3x_1 + 4x_2 \\ & 5x_1 + 6x_2 \geq 7 \\ & 8x_1 + 9x_2 \geq 10 \\ & 11x_1 + 12x_2 \geq 13 \\ & x_1, x_2 \geq 0 \end{array}$$

Dual

$$\begin{array}{ll} \max & 7u_1 + 10u_2 + 13u_3 \\ & 5u_1 + 8u_2 + 11u_3 \leq 3 \\ & 6u_1 + 9u_2 + 12u_3 \leq 4 \\ & u_1, u_2, u_3 \geq 0 \end{array}$$

In general:

$$\begin{array}{ll} \min & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{array} \qquad \begin{array}{ll} \max & b^\top u \\ & A^\top u \leq c \\ & u \geq 0 \end{array}$$

The **primal** has n variables and m constraints
$\Rightarrow$ The **dual** has m variables and n constraints

Properties of duality in LP

Weak duality: Given a primal $\min\{c^\top x : Ax \geq b, x \geq 0\}$ and its dual $\max\{b^\top u : A^\top u \leq c, u \geq 0\}$,

$$b^\top \bar{u} \leq c^\top \bar{x}$$

for any $\bar{x}$ and $\bar{u}$ feasible for their respective problems.

Strong duality: If a problem $\min\{c^\top x : Ax \geq b, x \geq 0\}$ is bounded and its dual $\max\{b^\top u : A^\top u \leq c, u \geq 0\}$ is bounded, their optimal solutions $\bar{x}$ and $\bar{u}$ coincide in value:

$$c^\top \bar{x} = b^\top \bar{u}$$

Properties of duality in LP (cont.)

Consequence: solving the dual or the primal doesn't matter: we get the same objective function value.

What if the primal (or the dual) is infeasible or unbounded?

Four cases:

- Primal bounded, dual bounded;
- Primal infeasible, dual infeasible;
- Primal unbounded ($c^\top x = -\infty$), dual infeasible;
- Primal infeasible, dual unbounded ($b^\top u = +\infty$).

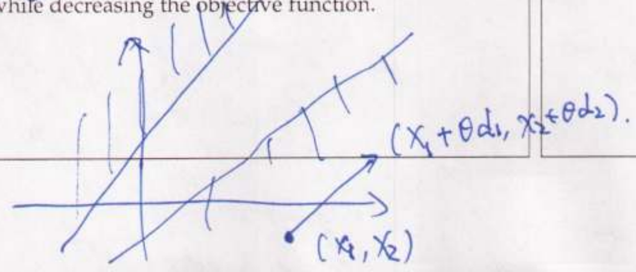
		Dual		
		bounded	unbounded	infeasible
Primal	bounded	Possible	–	–
	unbounded	–	–	Possible
	infeasible	–	Possible	Possible

Unbounded LP problem

Consider the following **minimization** problem:

$$\begin{array}{ll} \min & -5x_1 - 4x_2 \\ & 2x_1 - x_2 \geq 1 \\ & -x_1 + 2x_2 \geq 1 \\ & x_1, x_2 \geq 0 \end{array}$$

- How to show that the solution is unbounded?
- Consider direction (d_1, d_2) in which one can move infinitely while decreasing the objective function.



Unbounded LP problem $\Rightarrow$ Infeasible dual

to understand

$$\begin{cases} -5d_1 - 4d_2 < 0 \\ 2d_1 - d_2 \geq 0 \\ -d_1 + 2d_2 \geq 0 \\ d_1, d_2 \geq 0 \end{cases}$$

learn the way to consider problems

For example $(d_1, d_2) = (1, 1)$.

The dual

$$\begin{array}{ll} \max & u_1 + u_2 \\ & 2u_1 - u_2 \leq -5 \\ & -u_1 + 2u_2 \leq -4 \\ & u_1, u_2 \geq 0 \end{array}$$

$$\begin{array}{rl} 0 \leq & u_1(2d_1 - d_2) + u_2(-d_1 + 2d_2) \quad = \\ & d_1(2u_1 - u_2) + d_2(-u_1 + 2u_2) \leq \; -5d_1 - 4d_2 < 0 \end{array}$$

primal unboundedness - dual infeasibility

Primal - unbounded	Dual - infeasible
$\min\ c^\top x$ $Ax \ge b$ $x \ge 0$	$\max\ b^\top u$ $A^\top u \le c$ $u \ge 0$
$c^\top d < 0$ $Ad \ge 0$ $d \ge 0$	for any feasible u $0 \le d^\top A^\top u \le d^\top c < 0$ $u \ge 0$

Primal problem and dual problem with equality constraints

Primal	Dual
$\min\ 3x_1 + 4x_2$ $5x_1 + 6x_2 \ge 7$ $8x_1 + 9x_2 \ge 10$ $-8x_1 - 9x_2 \ge -10$ $11x_1 + 12x_2 \ge 13$ $x_1, x_2 \ge 0$	$\max\ 7u_1 + 10u_2' - 10u_2'' + 13u_3$ $5u_1 + 8u_2' - 8u_2'' + 11u_3 \le 3$ $6u_1 + 9u_2' - 9u_2'' + 12u_3 \le 4$ $u_1, u_2', u_2'', u_3 \ge 0$
$\min\ 3x_1 + 4x_2$ $5x_1 + 6x_2 \ge 7$ $8x_1 + 9x_2 = 10$ $11x_1 + 12x_2 \ge 13$ $x_1, x_2 \ge 0$	$\max\ 7u_1 + 10u_2 + 13u_3$ $5u_1 + 8u_2 + 11u_3 \le 3$ $6u_1 + 9u_2 + 12u_3 \le 4$ $u_1, u_3 \ge 0$, u_2 – unrstrtd

trick to reduce both sides of equations and unrstrtd variables

Primal problem and dual problem with unrestricted variables

Primal	Dual
$\min\ 3x_1 + 4(x_2' - x_2'')$ $5x_1 + 6(x_2' - x_2'') \ge 7$ $8x_1 + 9(x_2' - x_2'') \ge 10$ $11x_1 + 12(x_2' - x_2'') \ge 13$ $x_1, x_2', x_2'' \ge 0$	$\max\ 7u_1 + 10u_2 + 13u_3$ $5u_1 + 8u_2 + 11u_3 \le 3$ $6u_1 + 9u_2 + 12u_3 \le 4$ $-6u_1 - 9u_2 - 12u_3 \le -4$ $u_1, u_2, u_3 \ge 0$
$\min\ 3x_1 + 4x_2$ $5x_1 + 6x_2 \ge 7$ $8x_1 + 9x_2 \ge 10$ $11x_1 + 12x_2 \ge 13$ $x_1 \ge 0$, x_2 – unrstrtd	$\max\ 7u_1 + 10u_2 + 13u_3$ $5u_1 + 8u_2 + 11u_3 \le 3$ $6u_1 + 9u_2 + 12u_3 = 4$ $u_1, u_2, u_3 \ge 0$

↳ $x_2' - x_2''$ ($x_2', x_2'' \ge 0$)

How to construct the dual of an LP

Variable	Constraint
Constraint	Variable
Minimize	Maximize
Variable ≥ 0	Constraint $\le$
Variable ≤ 0	Constraint $\ge$
Var. Unrestricted	Constraint $=$
Constraint $\le$	Variable ≤ 0
Constraint $\ge$	Variable ≥ 0
Constraint $=$	Var. Unrestricted

LP primal and dual problem, standard form

Primal	Dual
$\min\ c^\top x$ $Ax = b$ $x \ge 0$	$\max\ b^\top u$ $A^\top u \le c$

$\min\ 5x_1 + 3x_2 - 2x_3$
$2x_1 + 4x_2 + x_3 = 4$
$-x_1 + x_2 + 2x_3 = 3$
$x_1, x_3, x_3 \ge 0$

$\max\ 4u_1 + 3u_2$
$$\begin{cases} 2u_1 - u_2 \le 5 \\ 4u_1 + u_2 \le 3 \\ u_1 + 2u_2 \le -2 \end{cases}$$

$$\Rightarrow \begin{cases} 2u_1 - u_2 + s_1 = 5 \\ 4u_1 + u_2 + s_2 = 3 \\ u_1 + 2u_2 + s_3 = -2 \end{cases} \quad s_1, s_2, s_3 \ge 0.$$

LP primal and dual problem, standard form

Primal	Dual
$\min\ c^\top x$ $Ax = b$ $x \ge 0$	$\max\ b^\top u$ $A^\top u + s = c$ $s \ge 0$

Slackness

What is the dual of the dual in standard form?

只是变成矩阵表达而已

$$\begin{array}{lll} \min\ c^\top x & & \max\ b^\top u & & \min\ c^\top x \\ \quad Ax = b & \rightarrow & \quad A^\top u + s = c & \rightarrow & \quad Ax = b \\ \quad x \geq 0 & & \quad s \geq 0 & & \quad Ix \geq 0 \\ & & \quad u - \text{unrstrtd} & & \quad x - \text{unrstrtd} \end{array}$$

S: slackness.

u: 是 constraint 变化对 objective 的影响
vector direction velocity.

S: constraint 的可变化量 (在objective上对对应xi的Ci).
这就与 shadow price 一致了. (unknown or difficult to calculate costs).

Complementary slackness

- Given a primal-dual pair, now we know how to solve one and get the optimal objective function of the other.

e.g. Solve primal $\Rightarrow$ get optimal obj.f. $c^\top \bar{x}$, an optimal solution $\bar{x}$, and the optimal dual obj.f. $b^\top \bar{u}$. **How do we get $\bar{u}$?**

> Complementary Slackness: If the primal problem
> $$\min\{c^\top x : \textstyle\sum_{i=1}^n a_{ji}x_i \geq b_j\ \forall j = 1, 2 \ldots, m, x \geq 0\}$$
> is bounded and admits optimum $\bar{x}$, and its dual
> $$\max\{b^\top u : \textstyle\sum_{j=1}^m a_{ji}u_j \leq c_i\ \forall i = 1, 2 \ldots, n, u \geq 0\}$$
> is bounded and admits optimal solution $\bar{u}$, then
> $$\bar{u}_j(\textstyle\sum_{i=1}^n a_{ji}\bar{x}_i - b_j) = 0 \quad \forall j = 1, 2 \ldots, m;$$
> $$\bar{x}_i(\textstyle\sum_{j=1}^m a_{ji}\bar{u}_j - c_i) = 0 \quad \forall i = 1, 2 \ldots, n$$

So if we solve the primal and get $\bar{x}$, we can get $\bar{u}$ by solving a system of equations.

LP primal and dual problem, standard form

$$\begin{array}{ll} \text{Primal} & \text{Dual} \\ \min\ c^\top x & \max\ b^\top u \\ \quad Ax = b & \quad A^\top u + s = c \\ \quad x \geq 0 & \quad s \geq 0 \end{array}$$

$$s_i x_i = 0 \Rightarrow (c_i - \sum_{j=1}^m a_{ji}u_j)x_i = 0$$

Either the primal variable is zero of the dual constraint is tight.

Example

$$\begin{array}{llll} \min & 3x_1 + 4x_2 & & \\ & 5x_1 + 6x_2 & \geq 7 \\ & 8x_1 + 9x_2 & \geq 10 \\ & 11x_1 + 12x_2 & \geq 13 \\ & x_1, x_2 \geq 0 & \end{array} \qquad \begin{array}{lll} \max & 7u_1 + 10u_2 + 13u_3 & \\ & 5u_1 + 8u_2 + 11u_3 & \leq 3 \\ & 6u_1 + 9u_2 + 12u_3 & \leq 4 \\ & u_1, u_2, u_3 \geq 0 & \end{array}$$

Solve the dual (with AMPL+CPLEX): get $(u_1, u_2, u_3) = (0.6, 0, 0)$.
Find (x_1, x_2) with complementary slackness:

$$\begin{array}{lcl} u_1(5x_1 + 6x_2 - 7) = 0 & & 0.6(5x_1 + 6x_2 - 7) = 0 \\ u_2(8x_1 + 9x_2 - 10) = 0 & & 0(8x_1 + 9x_2 - 10) = 0 \\ u_3(11x_1 + 12x_2 - 13) = 0 & \Rightarrow & 0(11x_1 + 12x_2 - 13) = 0 \\ x_1(5u_1 + 8u_2 + 11u_3 - 3) = 0 & & x_1(5 \cdot 0.6 + 8 \cdot 0 + 11 \cdot 0 - 3) = 0 \\ x_2(6u_1 + 9u_2 + 12u_3 - 4) = 0 & & x_2(6 \cdot 0.6 + 9 \cdot 0 + 12 \cdot 0 - 4) = 0 \end{array}$$

$$\begin{array}{lclcl} 5x_1 + 6x_2 = 7 & & 5x_1 + 6x_2 = 7 & & x_1 = \frac{7}{5} \\ x_1 \cdot 0 = 0 & \Rightarrow & x_1 \cdot 0 = 0 & \Rightarrow & x_2 = 0 \\ x_2 \cdot (-0.4) = 0 & & x_2 = 0 & & \end{array}$$

Another example

Consider the following LP problem:

$$\begin{array}{llllllll} \min & x_1 & +2x_2 & +3x_3 & -4x_4 & -3x_5 & \\ \text{s.t.} & -2x_1 & & -x_3 & -x_4 & & \geq 1 \\ & -x_1 & +x_2 & -x_3 & & +x_5 & \leq 2 \\ \multicolumn{8}{l}{x_1, x_2, x_3 \geq 0, x_4, x_5 \leq 0} \end{array}$$

1. Write its dual.
2. Solve the dual through the graphical method.
3. After finding the optimal value of the dual variables, use complementary slackness to find the optimal value of the primal variables.

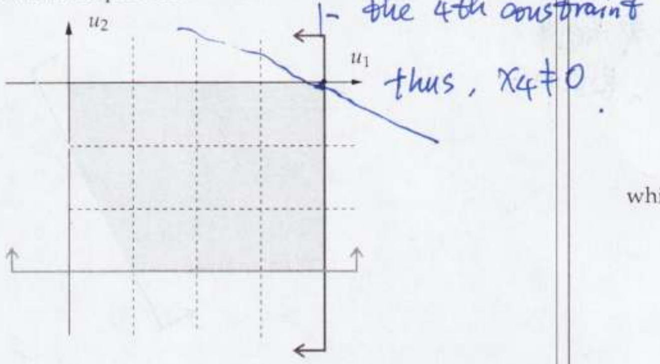
Another example: solution

The dual is

$$\begin{array}{llll} \max & u_1 & +2u_2 & \\ \text{s.t.} & -2u_1 & -u_2 & \leq 1 \\ & & u_2 & \leq 2 \\ & -u_1 & -u_2 & \leq 3 \\ & -u_1 & & \geq -4 \\ & & u_2 & \geq -3 \\ \multicolumn{4}{l}{u_1 \geq 0, u_2 \leq 0} \end{array}$$

Another example: solution

The second and third dual constraints are ignored as they are redundant to solve the problem.



The solution is clearly $(u_1, u_2) = (4, 0)$, corresponding to a value of 4 of the objective function.

Another example: solution

Complementarity slackness implies that:

$$
\begin{aligned}
&x_1(-2u_1 - u_2 - 1) = 0\\
&x_2(u_2 - 2) = 0\\
&x_3(-u_1 - u_2 - 3) = 0\\
&x_4(-u_1 + 4) = 0\\
&x_5(u_2 + 3) = 0\\
&u_1(-2x_1 - x_3 - x_4 - 1) = 0\\
&u_2(-x_1 + x_2 - x_3 + x_5 - 2) = 0
\end{aligned}
$$

which reduces, once we know the values of u_1 and u_2, to:

$$
\begin{aligned}
&x_1(-8 - 0 - 1) = 0\\
&x_2(0 - 2) = 0\\
&x_3(-4 - 0 - 3) = 0\\
&x_4(-4 + 4) = 0\\
&x_5(0 + 3) = 0\\
&-2x_1 - x_3 - x_4 = 1
\end{aligned}
$$

which implies $(x_1, x_2, x_3, x_5) = (0, 0, 0, 0)$, while $x_4 = -2 \cdot 0 - 0 - 1 = -1$.

Maximum Flow

$$
\begin{aligned}
\max \quad & \textstyle\sum_{j \in V:(j,t) \in A} x_{jt} \\
\text{s.t.} \quad & \textstyle\sum_{j \in V:(i,j) \in A} x_{ij} = \sum_{j \in V:(j,i) \in A} x_{ji} && \forall i \in V : s \neq i \neq t \\
& 0 \leq x_{ij} \leq c_{ij} && \forall (i,j) \in A
\end{aligned}
$$

- Variables for each node u_i for flow conservation constraints
- Variables for each arc z_{ij} for capacity constraints

$$
\begin{aligned}
\min \quad & \textstyle\sum_{(i,j) \in A} c_{ij} z_{ij} \\
\text{s.t.} \quad & z_{ji} \geq u_j - u_i && \forall (i,j) \in A,\ i \neq s\, j \neq t \\
& z_{si} \geq u_i && \forall (s,i) \in A \\
& z_{it} \geq 1 - u_{it} && \forall (i,t) \in A \\
& 0 \leq z_{ij} && \forall (i,j) \in A
\end{aligned}
$$

From complementarity slackness: $z_{ij}(x_{ij} - c_{ij}) = 0$.
What does it mean?

$\min\ 2z_{s1} + 3z_{12} + 2z_{2t} + \cancel{3z_{2t}}\ 3z_{s2} + 4z_{13} + z_{3t}$.

$u_1 + z_{s1} \geq 0$

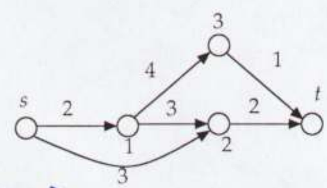
$u_2 + z_{s2} \geq 0$

$-u_1 + u_2 + z_{12} \geq 0$

$\vdots$

$-u_3 + z_{3t} \geq 1$.

Recall the Max-Flow example



$$
\begin{aligned}
\max \quad & x_{2t} + x_{3t} \\
& x_{s1} = x_{12} + x_{13} \\
& x_{s2} + x_{12} = x_{2t} \\
& x_{13} = x_{3t} \\
& 0 \leq x_{s1} \leq 2 \quad 0 \leq x_{s2} \leq 3 \\
& 0 \leq x_{12} \leq 3 \quad 0 \leq x_{13} \leq 4 \\
& 0 \leq x_{2t} \leq 2 \quad 0 \leq x_{3t} \leq 1
\end{aligned}
$$

z_{s1} z_{s2}

$x_{s1} - x_{12} - x_{13} = 0$ u_1

$\Rightarrow x_{s2} + x_{12} - x_{2t} = 0$ u_2

$x_{13} - x_{3t}$ u_3

$z_{ij}(x_{ij} - c_{ij}) = 0$.

used to find bound.

Transportation problem

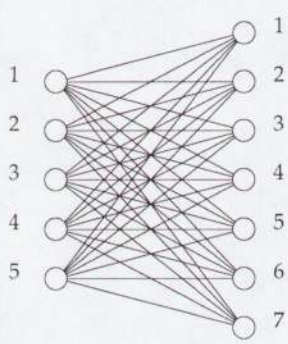
卖方市场

Variables: qty of product from producer $i \in P$ to distributor $j \in D$: x_{ij} (non-negative)

Constraints:

1. capacity: $\sum_{j \in D} x_{ij} \leq p_i \quad \forall i \in P$
2. demand: $\sum_{i \in P} x_{ij} \geq d_j \quad \forall j \in D$

Objective function: total transportation cost, $\min \sum_{i \in P} \sum_{j \in D} c_{ij} x_{ij}$



Dual of the transportation problem

买方市场.

- Variables for each supplier u_i for each supplier capacity constraints
- Variables for each distributer v_j for each distributer demand constraints

5 − 7

$$
\begin{aligned}
\max \quad & \textstyle\sum_{j \in D} d_j v_j - \sum_{i \in P} p_i u_i \\
\text{s.t.} \quad & v_j - u_i \leq c_{ij}\ \forall i \in P,\ j \in D \\
& 0 \leq u_i, v_j \qquad \forall i \in P,\ j \in D
\end{aligned}
$$

— 卖方行为

From complementarity slackness: $u_i(\sum_{j \in D} x_{ij} - p_i) = 0$ and $v_j(\sum_{i \in P} x_{ij} - d_j) = 0$.
From complementarity slackness: $x_{ij}(c_{ij} - v_j + u_i) = 0$.
What does it mean? Only send product from i to j if the difference between the "fair market" buy price for i and cell price for j equals the transportation cost.

Shadow prices

Consider an LP problem $\min\{c^T x : Ax \leq b\}$. Suppose we solved it to the optimum and an optimal solution is $x^\star$.

- associated with constraints u_i
- if nonzero, the constraint is active : for inequality $a^T x \leq b$, we have $a^T x^\star = b$ (equality constraints are always active)
- it can be interpreted as the "marginal value" of the constraint (or of the resource/budget/limit/... the constraint is associated with)

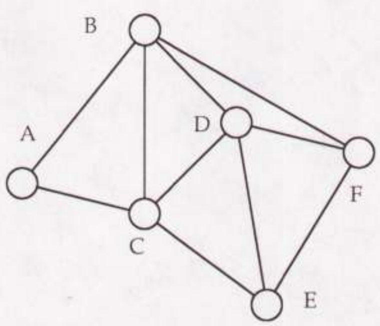
Reduced costs

- associated with variables
- if nonzero, the variable x_i is at its *lower* or *upper* bound
- gives an estimate of the "marginal value" of x_i
- i.e., if the coefficient of x_i in the objective function were lowered by that amount, the optimal solution would have $x_i \neq 0$.

Another example: the shortest path problem

For simplicity, the graph below is undirected, but we can assume for each edge there are two oppositely oriented arcs.

Suppose the problem is to compute the shortest path A → F.



The shortest path problem: primal

$$
\begin{array}{llll}
\min & c_{AB}x_{AB} + c_{BA}x_{BA} + \ldots + c_{EF}x_{EF} & + c_{FE}x_{FE} & \\
 & x_{AB} + x_{AC} & - x_{BA} - x_{CA} & = 1 \\
 & x_{BA} + x_{BC} + x_{BD} + x_{BF} & - x_{AB} - x_{CB} - x_{DB} - x_{FB} & = 0 \\
 & x_{CA} + x_{CB} + x_{CD} + x_{CE} & - x_{AC} - x_{BC} - x_{DC} - x_{EC} & = 0 \\
 & x_{DB} + x_{DC} + x_{DE} + x_{DF} & - x_{BD} - x_{CD} - x_{ED} - x_{FD} & = 0 \\
 & x_{EC} + x_{ED} + x_{EF} & - x_{CE} - x_{DE} - x_{FE} & = 0 \\
 & x_{FB} + x_{FD} + x_{FE} & - x_{BF} - x_{DF} - x_{DF} & = -1 \\
 & x_{AB}, x_{BA}, \ldots, x_{EF}, x_{FE} \geq 0 & &
\end{array}
$$

- We can express this as $\min\{c^\top x : Ax = b, x \geq 0\}$
- A is the adjacency matrix of G
- $|V|$ constraints, $|A|$ variables
- All constraints are equalities

The shortest path problem: dual

$$
\begin{array}{ll}
\max & u_A - u_F \\
 & u_A - u_B \leq c_{AB} \\
 & u_B - u_A \leq c_{BA} \\
 & u_A - u_C \leq c_{AC} \\
 & u_C - u_A \leq c_{CA} \\
 & \vdots \\
 & u_E - u_F \leq c_{EF} \\
 & u_F - u_E \leq c_{FE}
\end{array}
$$

- This is $\max\{b^\top u : A^\top u \leq c\}$
- $A^\top$ is the **transposed** adjacency matrix of G
- $|V|$ variables, $|A|$ constraints
- All variables are unrestricted in sign

what we should learn?

direction!!

ISE 426
Optimization models and applications

Lecture 10-11 — October 1, 2015

- basic feasible solutions
- simplex method
- connection with dual variables

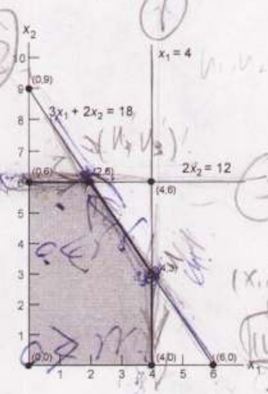
Reminders:
- Homework #2 is due in class 10/06. No late homework will be accepted!
- Quiz on 10/08, practice on 10/06.

Simplex Method

Example

maximize	$3x_1$	$+$	$5x_2$		
subject to	x_1			$\leq$	4
			$2x_2$	$\leq$	12
	$3x_1$	$+$	$2x_2$	$\leq$	18
	x_1			$\geq$	0
			x_2	$\geq$	0

The lines are the constraint boundaries.



(x_1, x_2)

(u_1, u_2, u_3)

$(x_1 = 4, x_2 = 3)$

$0 < W$

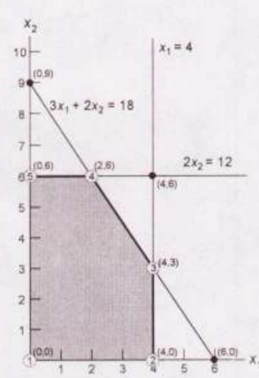
(3) ✓ $u_3(a_3x - b_3)$ $u_3 \neq 0$ u_3 u_1

(2) X $u_2(a_2x - b_2)$ $u_2 = 0.$

Corner-Point Solutions

5 corner-point feasible (CPF) solutions:

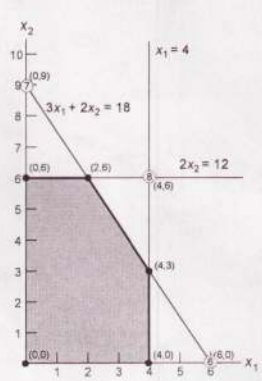
1. (0,0)
2. (4,0)
3. (4,3)
4. (2,6)
5. (0,6)



Corner-Point Solutions

3 corner-point infeasible solutions:

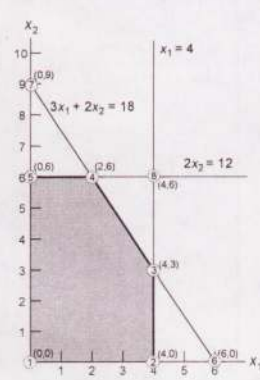
6. (6,0)
7. (0,9)
8. (4,6)



Corner-Point Solutions

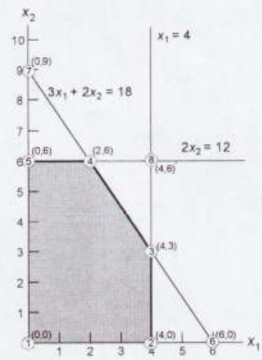
Each corner-point solution (feasible or infeasible) lies at the intersection of two constraint boundaries.

For an LP with n Decision Variables:
Each corner-point solution lies at the intersection of n constraint boundaries.



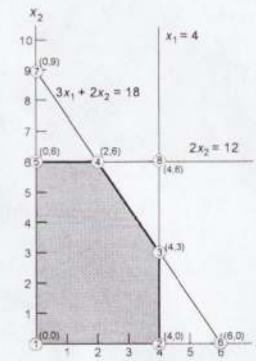
Adjacent CPFs

- For an LP with n decision variables, two CPF solutions are adjacent if they share $n-1$ constraint boundaries.
- Two adjacent CPF solutions are connected by an edge of the feasible region.



Adjacent CPFs

- For an LP with n decision variables, two CPF solutions are adjacent if they share $n-1$ constraint boundaries.
- Two adjacent CPF solutions are connected by an edge of the feasible region.
- (0,6) and (2,6) are adjacent.
- (2,6) and (4,3) are adjacent.

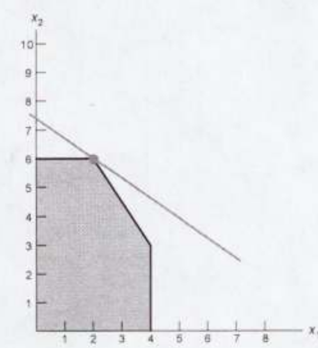


Why All the Fuss Over CPF Solutions?

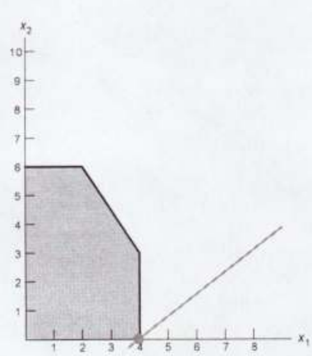
Important Property #1

If an LP has a single optimal solution, it is a CPF solution. If an LP has more than one optimal solution, at least two are CPF solutions.

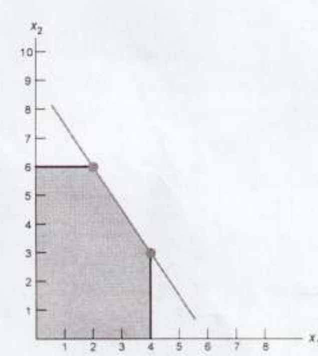
Why is it true?



Why is it true?



Why is it true?



How Many CPFs Are There?

- IP #1 means that we can focus only on CPF solutions and ignore the rest of the feasible region.
- There are an *infinite* number of feasible solutions
  - (assuming there are at least 2).
- There are a *finite* number of CPF solutions
  - (assuming feasible region is bounded and there are a finite number of constraints).
- That means we can focus on a *much smaller* set of possible answers.
- Can we just examine every CPF?

How Many CPFs? cont'd

- If there are n decision variables and m functional constraints, how many constraint boundaries are there?
- How many ways can we choose n constraint boundaries?
  - Answer:

$$\binom{m+n}{n} = \frac{(m+n)!}{m!n!}$$

- If $m = 50$ and $n = 50$, there are 10^{29} CPFs to examine
- If you could examine 1 billion CPFs per second, it would take you

$$3,170,979,198,376 \text{ years}$$

to examine all of the CPFs.

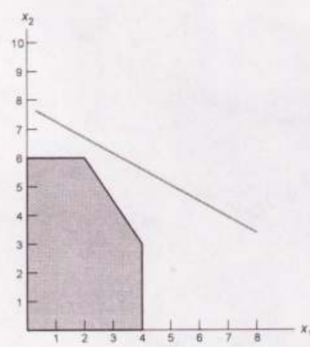
Why All the Fuss Over *Adjacent* CPF Solutions?

Important Property #2

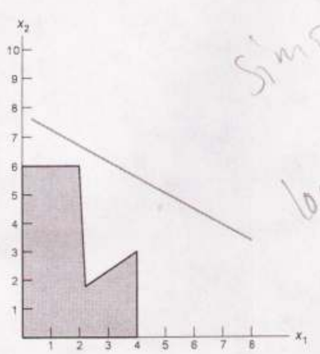
If a CPF solution has no *adjacent* CPF solutions that are better, then it must be an *optimal* solution.

In other words, if we find a CPF solution with no better neighbors, we can stop looking—there are no better solutions anywhere.

Why is this true?



Why is this true?



simplex

local optimal

convex.

global optimal

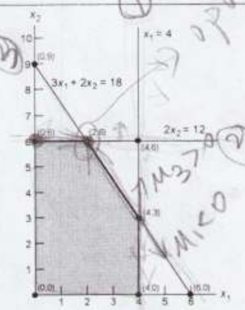
The Key Idea

Taken together, the two properties mean we can find an optimal solution by:

1. Starting at any CPF solution
2. Moving to a better adjacent CPF solution, if one exists
3. Continuing until the current CPF solution has no adjacent CPF solutions that are better

This is the essence of the simplex method.

The Dual



$$\begin{array}{llllllll} \text{minimize} & 4u_1 & + & 12u_2 & + & 18u_3 & & \\ \text{subject to} & u_1 & & & + & 3u_3 & \geq & 3 \\ & & & 2u_2 & + & 2u_3 & \geq & 5 \\ & u_1 & & & & & \geq & 0 \\ & & & u_2 & & & \geq & 0 \\ & & & & & u_3 & \geq & 0 \end{array}$$

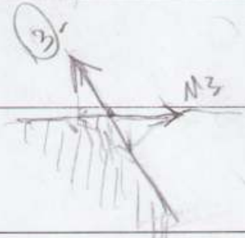
Example

$$\begin{array}{llllll} \max & 3x_1 & + & 5x_2 & & \\ \text{s.t.} & x_1 & & & \leq & 4 \\ & & & 2x_2 & \leq & 12 \\ & 3x_1 & + & 2x_2 & \leq & 18 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array} \qquad \begin{array}{llllllll} \min & 4u_1 & + & 12u_2 & + & 18u_3 & & \\ \text{s.t.} & u_1 & & & + & 3u_3 & \geq & 3 \\ & & & 2u_2 & + & 2u_3 & \geq & 5 \\ & u_1 & & & & & \geq & 0 \\ & & & u_2 & & & \geq & 0 \\ & & & & & u_3 & \geq & 0 \end{array}$$

Consider optimal $(x_1, x_2) = (2, 6)$, compute dual from complementary slackness:

$$\begin{array}{l} u_1(x_1 - 4) = 0 \\ u_2(+2x_2 - 12) = 0 \\ u_3(3x_1 + 2x_2 - 18) = 0 \\ x_1(u_1 + 3u_3 - 3) = 0 \\ x_2(2u_2 + 2u_3 - 5) = 0 \end{array} \quad \Rightarrow \quad \begin{array}{l} u_1(-2) = 0 \\ u_2(0) = 0 \\ u_3(0) = 0 \\ 2(u_1 + 3u_3 - 3) = 0 \\ 6(2u_2 + 2u_3 - 5) = 0 \end{array}$$

Example



$$\begin{array}{llllll} \max & 3x_1 & + & 5x_2 & & \\ \text{s.t.} & x_1 & & & \leq & 4 \\ & & & 2x_2 & \leq & 12 \\ & 3x_1 & + & 2x_2 & \leq & 18 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array} \qquad \begin{array}{llllllll} \min & 4u_1 & + & 12u_2 & + & 18u_3 & & \\ \text{s.t.} & u_1 & & & + & 3u_3 & \geq & 3 \\ & & & 2u_2 & + & 2u_3 & \geq & 5 \\ & u_1 & & & & & \geq & 0 \\ & & & u_2 & & & \geq & 0 \\ & & & & & u_3 & \geq & 0 \end{array}$$

Consider optimal $(x_1, x_2) = (2, 6)$, compute dual from complementary slackness:

$$\begin{array}{l} u_1 = 0 \\ 3u_3 = 3 \\ 2u_2 + 2u_3 = 5 \end{array} \quad \Rightarrow \quad \begin{array}{l} u_1 = 0 \\ u_2 = \frac{3}{2} \\ u_3 = 1 \end{array}$$

Example

$$\begin{array}{llllll} \max & 3x_1 & + & 5x_2 & & \\ \text{s.t.} & x_1 & & & \leq & 4 \\ & & & 2x_2 & \leq & 12 \\ & 3x_1 & + & 2x_2 & \leq & 18 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array} \qquad \begin{array}{llllllll} \min & 4u_1 & + & 12u_2 & + & 18u_3 & & \\ \text{s.t.} & u_1 & & & + & 3u_3 & \geq & 3 \\ & & & 2u_2 & + & 2u_3 & \geq & 5 \\ & u_1 & & & & & \geq & 0 \\ & & & u_2 & & & \geq & 0 \\ & & & & & u_3 & \geq & 0 \end{array}$$

Consider a feasible CPF $(x_1, x_2) = (4, 3)$, compute dual from complementary slackness:

$$\begin{array}{l} u_1(x_1 - 4) = 0 \\ u_2(+2x_2 - 12) = 0 \\ u_3(3x_1 + 2x_2 - 18) = 0 \\ x_1(u_1 + 3u_3 - 3) = 0 \\ x_2(2u_2 + 2u_3 - 5) = 0 \end{array} \quad \Rightarrow \quad \begin{array}{l} u_1(0) = 0 \\ u_2(-4) = 0 \\ u_3(0) = 0 \\ 2(u_1 + 3u_3 - 3) = 0 \\ 6(2u_2 + 2u_3 - 5) = 0 \end{array}$$

Example

$$\begin{array}{llllll} \max & 3x_1 & + & 5x_2 & & \\ \text{s.t.} & x_1 & & & \leq & 4 \\ & & & 2x_2 & \leq & 12 \\ & 3x_1 & + & 2x_2 & \leq & 18 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array} \qquad \begin{array}{lllllllll} \min & 4u_1 & + & 12u_2 & + & 18u_3 & & & \\ \text{s.t.} & u_1 & & & + & 3u_3 & \geq & 3 & x_1 \\ & & & 2u_2 & + & 2u_3 & \geq & 5 & x_2 \\ & u_1 & & & & & \geq & 0 & \\ & & & u_2 & & & \geq & 0 & \\ & & & & & u_3 & \geq & 0 & \end{array}$$

Consider a feasible CPF $(x_1, x_2) = (4, 3)$, compute dual from complementary slackness:

$$\begin{array}{l} u_2 = 0 \\ u_1 + 3u_3 = 3 \\ 2u_3 = 5 \end{array} \quad \Rightarrow \quad \begin{array}{l} u_1 = -\frac{9}{2} \\ u_2 = 0 \\ u_3 = \frac{5}{2} \end{array}$$

$s_1 = -\frac{9}{2} + \frac{5}{2} \times 3 - 3 = 0$

$s_2 = 2 \times \frac{5}{2} - 5 = 0$

Example

$$\begin{array}{llllll} \max & 3x_1 & + & 5x_2 & & \\ \text{s.t.} & x_1 & & & \leq & 4 \\ & & & 2x_2 & \leq & 12 \\ & 3x_1 & + & 2x_2 & \leq & 18 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array} \qquad \begin{array}{llllllll} \min & 4u_1 & + & 12u_2 & + & 18u_3 & & \\ \text{s.t.} & u_1 & & & + & 3u_3 & \geq & 3 \\ & & & 2u_2 & + & 2u_3 & \geq & 5 \\ & u_1 & & & & & \geq & 0 \\ & & & u_2 & & & \geq & 0 \\ & & & & & u_3 & \geq & 0 \end{array}$$

Consider a feasible CPF $(x_1, x_2) = (0, 6)$, compute dual complementary solution:

$$\begin{array}{l} u_1(x_1 - 4) = 0 \\ u_2(+2x_2 - 12) = 0 \\ u_3(3x_1 + 2x_2 - 18) = 0 \\ x_1(u_1 + 3u_3 - 3) = 0 \\ x_2(2u_2 + 2u_3 - 5) = 0 \end{array} \quad \Rightarrow \quad \begin{array}{l} u_1(-4) = 0 \\ u_2(0) = 0 \\ u_3(-6) = 0 \\ 0(u_1 + 3u_3 - 3) = 0 \\ 6(2u_2 + 2u_3 - 5) = 0 \end{array}$$

对应 optimal corner，用 complementary ~~solution~~ slackness 解出来的 duality - u_i 太 infeasible. { ① s.t. ⇒ $u<0$ ② variable $\lessgtr 0$ ⇒ slackness.

Example

$$\begin{array}{llcll} \max & 3x_1 & + & 5x_2 & \\ \text{s.t.} & x_1 & & & \leq 4 \\ & & & 2x_2 & \leq 12 \\ & 3x_1 & + & 2x_2 & \leq 18 \\ & x_1 & & & \geq 0 \\ & & & x_2 & \geq 0 \end{array} \qquad \begin{array}{llllll} \min & 4u_1 & + 12u_2 & + 18u_3 & & \\ \text{s.t.} & u_1 & & + 3u_3 & \geq & 3 \\ & & 2u_2 & + 2u_3 & \geq & 5 \\ & u_1 & & & \geq & 0 \\ & & u_2 & & \geq & 0 \\ & & & u_3 & \geq & 0 \end{array}$$

Consider a feasible CPF $(x_1, x_2) = (0, 6)$, compute dual complementary solution:

$$\begin{array}{l} u_1 = 0 \\ 2u_2 = 5 \\ u_3 = 0 \end{array} \qquad \Rightarrow \begin{array}{l} u_1 = 0 \\ u_2 = \frac{5}{2} \\ u_3 = 0 \\ u_1 + 3u_3 - 3 = -3 \end{array}$$

i: ~~(≥0)~~ variable + constraint
[about 0] consider S_i. (x_i 对应)

Example

$$\begin{array}{llcll} \max & 3x_1 & + & 5x_2 & \\ \text{s.t.} & x_1 & & & \leq 4 \\ & & & 2x_2 & \leq 12 \\ & 3x_1 & + & 2x_2 & \leq 18 \\ & x_1 & & & \geq 0 \\ & & & x_2 & \geq 0 \end{array} \qquad \begin{array}{llllll} \min & 4u_1 & + 12u_2 & + 18u_3 & & \\ \text{s.t.} & u_1 & & + 3u_3 & \geq & 3 \\ & & 2u_2 & + 2u_3 & \geq & 5 \\ & u_1 & & & \geq & 0 \\ & & u_2 & & \geq & 0 \\ & & & u_3 & \geq & 0 \end{array}$$

Consider a feasible CPF $(x_1, x_2) = (0, 6)$, compute dual complementary solution:

- $s_1 = u_1 + 3u_3 - 3 = -3$ is the reduced cost for the variable $x_1 \geq 0$ at the solution $x_1 = 0$. opportunity cost
- $s_1 < 0$ means that the objective function will improve if we allow $x_1 > 0$ while we keep $2x_2 = 12$.
- Consider $x_1 = 1$, then $x_2 = 6$ then $3x_1 + 5x_2 = 3 * 0 + 5 * 6 + 3$.

Changes in Objective Function Coefficients

- Suppose x^* is the optimal solution for an LP.
- Z^* is its optimal objective value.
- Suppose that some objective function coefficient c_j changes.

$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \leq 4 \\ \quad 2x_2 & \leq 12 \\ 3x_1 + 2x_2 & \leq 18 \\ x_1 \ , \ x_2 & \geq 0 \end{array}$$

- What if the 3 changed?

Changes in c, cont'd

$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \leq 4 \\ \quad 2x_2 & \leq 12 \\ 3x_1 + 2x_2 & \leq 18 \\ x_1 \ , \ x_2 & \geq 0 \end{array}$$

- If 3 changed to 0, what would the new solution be?
- If 3 changed to 30, what would the new solution be?
- If the 3 changed to $3 \pm \delta$, where δ is tiny, what would the new solution be?

Example: Maximization

$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \leq 4 \\ \quad 2x_2 & \leq 12 \\ 3x_1 + 2x_2 & \leq 18 \\ x_1 \ , \ x_2 & \geq 0 \end{array}$$

- Optimal solution is $(x_1^*, x_2^*) = (2, 6)$, $Z^* = 36$.
- Suppose the 3 increased to 7.
- Which of the following is true?
  1. Z^* will increase.
  2. Z^* will decrease.
  3. Z^* will stay the same.
  4. We can't say.

Example: Maximization, cont'd

$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \leq 4 \\ \quad 2x_2 & \leq 12 \\ 3x_1 + 2x_2 & \leq 18 \\ x_1 \ , \ x_2 & \geq 0 \end{array}$$

- $(x_1^*, x_2^*) = (2, 6)$, $Z^* = 36$.
- 3 increases to 7.
- The local rate of increase is derived from $x_1^* \delta = 2 \times 4 = 8$.
- Which of the following is true?
  1. Z^* will increase by *exactly* 8.
  2. Z^* will increase by *at most* 8.
  3. Z^* will increase by *at least* 8.
  4. Z^* will increase, but we don't know by how much.

(x_1^*, x_2^*) 不变时 · $=8$.
变3，故 >8.

Changes in Constraint Right-Hand Sides

$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \le 4 \\ \quad\quad 2x_2 & \le 12 \\ \quad 3x_1 + 2x_2 & \le 18 \\ \quad x_1\ ,\ x_2 & \ge 0 \end{array}$$

- Suppose 12 increased.
- Would optimal solution change?
- Would optimal objective value change?
- Would optimal basis change?

Example

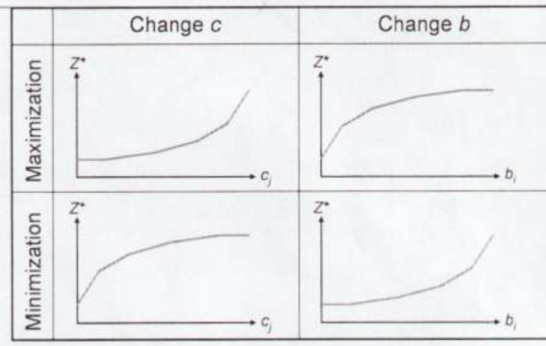
$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \le 4 \\ \quad\quad 2x_2 & \le 12 \\ \quad 3x_1 + 2x_2 & \le 18 \\ \quad x_1\ ,\ x_2 & \ge 0 \end{array}$$

- Optimal solutions are $(x_1^*, x_2^*) = (2, 6)$, $(u_1^*, u_2^*, u_3^*) = (0, \frac{3}{2}, 1)$, $Z^* = 36$.
- Suppose the 12 increased to 16.
- Which of the following is true?
  1. Z^* will increase.
  2. Z^* will decrease.
  3. Z^* will stay the same.
  4. We can't say.

Example, cont'd

$$\begin{array}{ll} \max Z = 3x_1 + 5x_2 & \\ \text{s.t.} \quad x_1 & \le 4 \\ \quad\quad 2x_2 & \le 12 \\ \quad 3x_1 + 2x_2 & \le 18 \\ \quad x_1\ ,\ x_2 & \ge 0 \end{array}$$

- $(x_1^*, x_2^*) = (2, 6)$, $(u_1^*, u_2^*, u_3^*) = (0, \frac{3}{2}, 1)$, $Z^* = 36$. $(\frac{2}{3}, 8)$
- 12 increases to 16.
- The local rate of increase come from the shadow price: $u_2^*\delta = \frac{3}{2} \times 4 = 6$. $2 + 40 = 42$ $-36 = 6$
- Which of the following is true?
  1. Z^* will increase by *exactly* 6.
  2. Z^* will increase by *at most* 6.
  3. Z^* will increase by *at least* 6.
  4. Z^* will increase, but we don't know by how much.



Relationship to Complementary Slackness

不紧的 constraints.

Right-Hand Sides

only change slackness

- Suppose there is slack in the ith primal constraint.
  - Increasing the RHS would not change the optimal solution.
  - By complementary slackness, u_i^* must equal 0 (in the dual).
  - Using the statement on the previous slides, the optimal objective function changes by u_i^*, or 0.
- Suppose there is no slack in the ith primal constraint.
  - Increasing the RHS *would* change the optimal solution.
  - u_i^* *probably* (!) is greater than 0.
  - Using the statement above, the optimal objective function changes by u_i^*.
- This agrees with our interpretation of the dual values as *shadow prices*.

Relationship to Complementary Slackness, cont'd

- Now suppose there is slack in the jth dual constraint.
  - By complementary slackness, $x_j^* = 0$ (in the primal).
  - If we increase c_j slightly, we'll still want to set $x_j^* = 0$.
  - We argued that for each unit increase in c_j, Z^* changes by x_j^* (if optimal basis stays the same).
  - So Z^* increases by 0 when c_j increases.
- Suppose there is no slack in the jth dual constraint (reduced cost is 0).
  - $x_j^* > 0$ (probably).
  - If we increase c_j by 1, the objective value will go up by x_j^*.

primal
$$x_1 \le 4$$
$$2x_2 \le 12$$
$$3x_1 + 2x_2 \le 18$$

Slackness :
$(x_1-4)\mu_1=0$ $\qquad x_1(\mu_1+3\mu_3-3)=0$

$(x_2-6)\mu_2=0$ $\qquad x_2(2\mu_1+2\mu_3-5)=0.$

$(3x_1+2x_2-18)\mu_3=0$

Example : $(x_1, x_2) = (4, 3).$

$$\begin{cases} \mu_1 = -\frac{9}{2} \\ \mu_2 = 0 \\ \mu_3 = \frac{5}{2} \end{cases}$$

① let $x_1 \le 4$ changes to $x_1 \le 5$.

Then: $\underline{(4,3)} \Rightarrow \underline{(5, \frac{3}{2})}$

$z_0 = 3\times4+5\times3 = 27$ $\qquad z' = 5\times3+\frac{3}{2}\times5 = \frac{45}{2}$

constraint 改变一单位对 optimal 的影响.

$\frac{45}{2} - 27 = -\frac{9}{2} = \mu_1$

② let $3x_1+2x_2 \le 18$ to $3x_1+2x_2 \le 19$

Then : $(4,3) \Rightarrow (4, \frac{7}{2})$

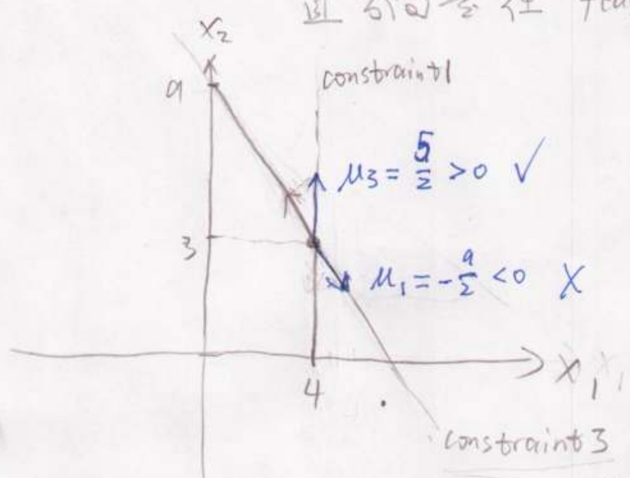
$z_0 = 3\times4+5\times3 = 27,$ $\qquad z'' = 4\times3+\frac{7}{2}\times5 = \frac{59}{2}.$

$\frac{59}{2} - \frac{54}{2} = \frac{5}{2} = \mu_3$

∴ μ_i 的方向并不是沿 constraint i 的

而是沿与它相邻的 constraint j 的.

且方向是往 feasible set 外。



因为沿 μ_3 方向 >0,

μ_3 在 constraint 3 上投影也 >0,

μ_1 相反

故沿 constraint 3 方向走.

在多维中，我们找 optimal corner 的思路应该是排除 $\mu<0$ 的 constraint. when a $\mu<0$, remove that constraint, change to a new one. 然后看 slackness.

ISE 426

Optimization models and applications

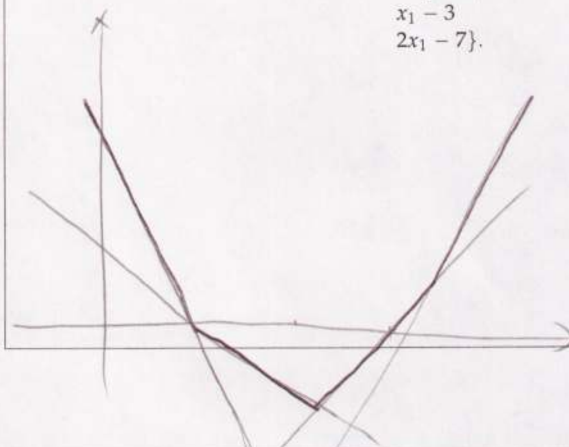
Lecture 13 — October 15, 2015

- MinMax
- Goal programming
- Winston & Venkataramanan, pages 191-194

Minimizing the maximum of a set of linear functions

Consider an optimization problem of the form

$$\min \ \max\{ \ -2x_1+2,\ -x_1+1,\ x_1-3\ 2x_1-7\}.$$



Minimizing the maximum of a set of linear functions

Consider an optimization problem of the form

$$\min \ \max\{ \ -2x_1+2,\ -x_1+1,\ x_1-3\ 2x_1-7\}.$$

- This is convex nonlinear problem
- How to reformulate it into a linear problem
- Create a new variable y
- y is the **maximum** of all quantities $-2x_1+2$, $-x_1+1$, x_1-3, $2x_1-7$.

Minimizing the maximum of a set of linear functions

In general, consider an optimization problem of the form

$$\min \ \max_{k=1,2\ldots,H} \ (\textstyle\sum_{j=1}^n a_{kj}x_j + b_k) =$$

$$\min \ \max \ \{\textstyle\sum_{j=1}^n a_{1j}x_j + b_1,\ \sum_{j=1}^n a_{2j}x_j + b_2,\ \vdots\ \sum_{j=1}^n a_{Hj}x_j + b_H\}.$$

- This is nonlinear (there's a max term in the objective)[1].

min $\{max\{-2x_1+2, -x_1+1\} + max\{x_1+3, 2x_1-7\}\}$

min $y+z$ $\quad y \geq -2x_1+2$, $y \geq -x_1+1$

$z \geq x_1+3$, $z \geq 2x_1-7$

[1]AMPL won't complain, but CPLEX will refuse to solve the problem.

Minimizing the maximum of a set of linear functions

In general, consider an optimization problem of the form

$$\min \ \max_{k=1,2\ldots,H} \ (\textstyle\sum_{j=1}^n a_{kj}x_j + b_k) =$$

$$\min \ \max \ \{\textstyle\sum_{j=1}^n a_{1j}x_j + b_1,\ \sum_{j=1}^n a_{2j}x_j + b_2,\ \vdots\ \sum_{j=1}^n a_{Hj}x_j + b_H\}.$$

- This is nonlinear (there's a max term in the objective)[1].
- However, the model easily becomes linear:
- Create a new variable y
- y is the **maximum** of all quantities $\sum_{j=1}^n a_{1j}x_j + b_1$, $\sum_{j=1}^n a_{2j}x_j + b_2 \ldots$, $\sum_{j=1}^n a_{Hj}x_j + b_H$.

[1]AMPL won't complain, but CPLEX will refuse to solve the problem.

Minimizing the maximum of a set of linear functions

- Easy… if y is the maximum of all those quantities, then it must be greater than each of them:

$$\begin{aligned} y &\geq \textstyle\sum_{j=1}^n a_{1j}x_j + b_1, \\ y &\geq \textstyle\sum_{j=1}^n a_{2j}x_j + b_2, \\ &\vdots \\ y &\geq \textstyle\sum_{j=1}^n a_{Hj}x_j + b_H. \end{aligned}$$

- each of these constraints is linear!
  (re-write as $\sum_{j=1}^n a_{1j}x_j - y \leq -b_1, \ldots$)
- objective function is y. The linear model is:

$$\begin{aligned} \min \quad & y \\ & y \geq \textstyle\sum_{j=1}^n a_{kj}x_j + b_k \qquad \forall k = 1,2\ldots,H \end{aligned}$$

Why do we minimize y?

- The constraints above only say that y is **at least** the maximum of all those linear functions.

$\Rightarrow$ They don't guarantee that y is **exactly** the maximum of all those linear functions.

- That is,

$$y \geq \sum_{j=1}^{n} a_{kj}x_j + b_k \qquad \forall k = 1, 2 \ldots, H$$

only ensures that

$$y \;\geq\; \max_{k=1,2,\ldots,H} \textstyle\sum_{j=1}^{n} a_{kj}x_j + b_k.$$

It does **not** ensure that

$$y \;=\; \max_{k=1,2,\ldots,H} \textstyle\sum_{j=1}^{n} a_{kj}x_j + b_k.$$

Minimizing the maximum of a set of functions

However, this model works as we are minimizing y:

- Although for all feasible solutions $y \geq \max_{k=1,2,\ldots,H} \sum_{j=1}^{n} a_{kj}x_j + b_k$,
- a solution $(\bar{x}_1, \bar{x}_2 \ldots, \bar{x}_n, \bar{y})$ with

$$\bar{y} > \max_{k=1,2,\ldots,H} \sum_{j=1}^{n} a_{kj}\bar{x}_j + b_k$$

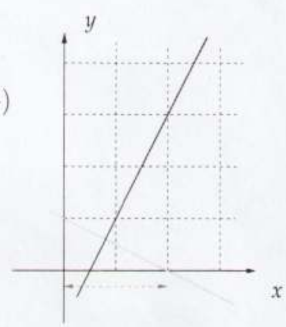
(strictly $>$) is feasible, but not optimal.

- Question: for an optimal solution $(\bar{x}_1, \bar{x}_2 \ldots, \bar{x}_n, \bar{y})$ for how many k's $\bar{y} = \sum_{j=1}^{n} a_{kj}\bar{x}_j + b_k$?

Another example

$$\begin{array}{ll} \min & \left(\max\{2x-1,\ -\tfrac{1}{2}x+1\}\right) \\ & 0 \leq x \leq 2 \end{array}$$

$$\begin{array}{ll} \min & y \\ & y \geq 2x-1 \\ & y \geq -\tfrac{1}{2}x+1 \\ & 0 \leq x \leq 2 \end{array}$$

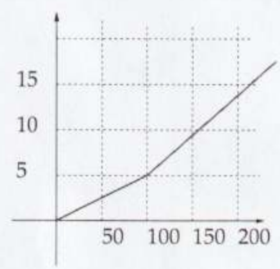


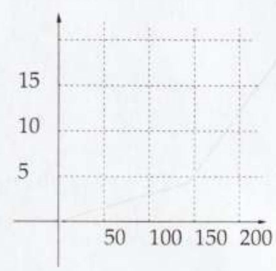
Example: bank loan

Our company wants to ask for a loan of 300k$ to two banks. The interests paid to a bank depend on amount borrowed:

Bank 1:
- 5% of amt $\leq$ 100k$
- 8% of amt $\geq$ 100k$

Bank 2:
- 3% of amt $\leq$ 140k$
- 12% of amt $\geq$ 140k$





Example: bank loan

Determine how much to borrow from both banks in order to minimize the total interests paid.

- Variables: x_1 and x_2, amount borrowed from Bank 1 and Bank 2 (in k$)
- Constraints: $x_1 \geq 0$, $x_2 \geq 0$, and $x_1 + x_2 = 300$
- Objective function: sum of interests paid to the banks.

$$\min f_1(x_1) + f_2(x_2)$$

What are $f_1(x_1)$ and $f_2(x_2)$?

$f_1(x_1)$: $0.05 * x_1$ for $0 \leq x_1 \leq 100$,
$5 + 0.08 * (x_1 - 100)$ for $x_1 \geq 100$

$f_2(x_2)$: $0.03 * x_2$ for $0 \leq x_2 \leq 140$,
$4.2 + 0.12 * (x_2 - 140)$ for $x_2 \geq 140$

Example: bank loan

For this specific case[2], both $f_1(x_1)$ and $f_2(x_2)$ can be written as

$$f_1(x_1) = \max\{0.05 * x_1,\ 5 + 0.08 * (x_1 - 100)\}$$
$$f_2(x_2) = \max\{0.03 * x_2,\ 4.2 + 0.12 * (x_2 - 140)\}$$

So the model is:

$$\begin{array}{rl} \min & \max\{0.05 * x_1,\ 5 + 0.08 * (x_1 - 100)\} + \\ + & \max\{0.03 * x_2,\ 4.2 + 0.12 * (x_2 - 140)\} \\ & \\ & x_1 + x_2 = 300 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{array}$$

Nonlinear...

[2] both f_1 and f_2 are convex!

Example: bank loan

Linearization:

$$
\begin{aligned}
\min\quad & y_1 + y_2 \\
& y_1 \geq 0.05 * x_1 \\
& y_1 \geq 5 + 0.08 * (x_1 - 100) \\
& y_2 \geq 0.03 * x_2 \\
& y_2 \geq 4.2 + 0.12 * (x_2 - 140) \\
& x_1 + x_2 = 300 \\
& x_1 \geq 0 \\
& x_2 \geq 0
\end{aligned}
$$

Attaining minimum

- the auxillary variable "wants to" be at its lowest allowed value
- it should appear with a positive coefficient in a minimization problem or with a negative one in a max. problem

Symmetrically, it also works in "max-min" problems, that is, when maximizing the minimum of a set of functions
Caution! It does **not** work in general in other contexts, e.g.:

$$\max \max_{k=1,2\dots,H}\left(\sum_{j=1}^{n} a_{kj}x_j + b_k\right)$$

or

$$\min \min_{k=1,2\dots,H}\left(\sum_{j=1}^{n} a_{kj}x_j + b_k\right)$$

Example: job assignment

Problem:

- We have to assign m workers to m jobs. Everyone must be assigned to exactly one job, and all jobs have to be done.
- The degree of preference of a worker i to job j is defined by c_{ij}, for $i = 1, 2 \dots, m$, $j = 1, 2 \dots, m$.
- maximize the total preference, i.e. the sum of all preferences c_{ij} for assignments (i, j) worker-job.

Job assignment: model

Variables: x_{ij} for worker i and job j. Constraints:

- Every worker is assigned to **exactly** one job:

$$\sum_{j=1}^{m} x_{ij} = 1 \quad \forall i = 1, 2 \dots, m$$

- Every job is done by **exactly** one worker:

$$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j = 1, 2 \dots, m$$

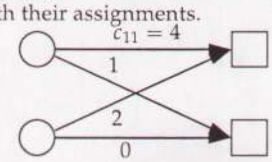
- Variables x_{ij} are binary (a yes/no decision)

Objective function: total preference

$$\sum_{i=1}^{m}\sum_{i=1}^{m} c_{ij}x_{ij}$$

Bad objective?

- the total preference $\sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij}x_{ij}$ does not provide a fair balance in assigning jobs: some worker may be very dissatisfied with their assignments.



fair

- For a fair assignment, we may instead maximize the minimum assignment cost of each worker:
- How? The satisfaction of worker i is equal to $\sum_{j=1}^{m} c_{ij}x_{ij}$
- New objective function (still to be maximized):

$$\min_{i=1,2\dots,m} \sum_{j=1}^{m} c_{ij}x_{ij}$$

- Look at least satisfied worker(s) (as it results from variables x_{ij}) and limit their dissatisfaction as much as possible

Job assignment: new model

$$
\begin{array}{lll}
\max & \min_{i=1,2\dots,m} \sum_{j=1}^{m} c_{ij}x_{ij} & \\
& \sum_{j=1}^{m} x_{ij} = 1 & \forall i = 1, 2 \dots, m \\
& \sum_{i=1}^{m} x_{ij} = 1 & \forall j = 1, 2 \dots, m \\
& x_{ij} \in \{0, 1\} & \forall i, j = 1, 2 \dots, m
\end{array}
$$

It's nonlinear! Let's use the same trick, with different signs.

- New variable y (will be our objective function)
- y is $\min_{i=1,2\dots,m} \sum_{j=1}^{m} c_{ij}x_{ij}$, for each $i = 1, 2 \dots, m$.

$\Rightarrow$ y is smaller than each of these quantities:

$$
\begin{aligned}
y &\leq \textstyle\sum_{j=1}^{m} c_{1j}x_{1j} \\
y &\leq \textstyle\sum_{j=1}^{m} c_{2j}x_{2j} \\
&\vdots \\
y &\leq \textstyle\sum_{j=1}^{m} c_{mj}x_{mj}
\end{aligned}
$$

- or, more compact: $y \leq \sum_{j=1}^{m} c_{ij}x_{ij}\ \forall i = 1, 2 \dots, m$

Job assignment: final linear model

$$
\begin{array}{lll}
\max & y & \\
& y \leq \sum_{j=1}^{m} c_{ij}x_{ij} & \forall i = 1,2\dots,m \\
& \sum_{j=1}^{m} x_{ij} = 1 & \forall i = 1,2\dots,m \\
& \sum_{i=1}^{m} x_{ij} = 1 & \forall j = 1,2\dots,m \\
& x_{ij} \in \{0,1\} & \forall i,j = 1,2\dots,m
\end{array}
$$

Job assignment: alternative model

Let's reduce it to a minimization problem. The obj.f. changes sign, the problem becomes a minimization one:

$$
\begin{aligned}
& \max \quad \min_{i=1,2\dots,m} \sum_{j=1}^{m} c_{ij}x_{ij} = \\
= & -\min \left(-\min_{i=1,2\dots,m} \sum_{j=1}^{m} c_{ij}x_{ij} \right) =
\end{aligned}
$$

[apply the inverse rule inside the brackets...]

$$
\begin{aligned}
= & -\min \left(\max_{i=1,2\dots,m} \left(-\sum_{j=1}^{m} c_{ij}x_{ij} \right) \right) = \\
= & -\min \left(\max_{i=1,2\dots,m} \sum_{j=1}^{m} (-c_{ij})x_{ij} \right)
\end{aligned}
$$

Minimizing the maximum of absolute values

Consider now a system, of linear equations $Ax = b$, which does not have a solution. We want to find a solution x which violates the linear quations as little as possible. We can consider

$$
\begin{array}{lll}
\min & \max_{k=1,2\dots,H} & (|\sum_{j=1}^{n} a_{kj}x_j - b_k| = \\
\min & \max & \{|\sum_{j=1}^{n} a_{1j}x_j - b_1|, \\
& & |\sum_{j=1}^{n} a_{2j}x_j - b_2|, \\
& & \vdots \\
& & |\sum_{j=1}^{n} a_{Hj}x_j - b_H|\}.
\end{array}
$$

make max slackness be min

Minimizing the maximum of absolute values

Consider now a system, of linear equations $Ax = b$, which does not have a solution. We want to find a solution x which violates the linear quations as little as possible. We can consider

$$
\begin{array}{lll}
\min & \max_{k=1,2\dots,H} & (|\sum_{j=1}^{n} a_{kj}x_j - b_k| = \\
\min & \max & \{|\sum_{j=1}^{n} a_{1j}x_j - b_1|, \\
& & |\sum_{j=1}^{n} a_{2j}x_j - b_2|, \\
& & \vdots \\
& & |\sum_{j=1}^{n} a_{Hj}x_j - b_H|\}.
\end{array}
$$

$$
\begin{array}{lll}
\min & y & \\
& y \geq \sum_{j=1}^{n} a_{kj}x_j - b_k & \forall k = 1,2\dots,H \\
& y \geq -(\sum_{j=1}^{n} a_{kj}x_j - b_k) & \forall k = 1,2\dots,H
\end{array}
$$

The solution is likely to have n of the constraints from $Ax = b$ to have equal maximum violation. What if we do not like such a solution?

Minimizing the sum of absolute values

Consider now a system, of linear equations $Ax = b$, which does not have a solution. We want to find a solution x which violates the linear equations as little as possible in total. We can consider

$$
\begin{array}{ll}
\min & \sum_{k=1}^{H}(|\sum_{j=1}^{n} a_{kj}x_j - b_k| = \\
\min & \sum_{k=1}^{H} |y_i| \\
& y_1 = \sum_{j=1}^{n} a_{1j}x_j - b_1, \\
& y_2 = \sum_{j=1}^{n} a_{2j}x_j - b_2, \\
& \vdots \\
& y_H = \sum_{j=1}^{n} a_{Hj}x_j - b_H.
\end{array}
$$

[illegible] ∵ min all $(y'_k + y''_k)$ and y'_k or $y''_k = 0$.

Minimizing the sum of absolute values

Consider now a system, of linear equations $Ax = b$, which does not have a solution. We want to find a solution x which violates the linear equations as little as possible in total. We can consider

$$
\begin{array}{ll}
\min & \sum_{k=1}^{H}(|\sum_{j=1}^{n} a_{kj}x_j - b_k| = \min \sum_{k=1}^{H} y_i \\
\min & \sum_{k=1}^{H} |y_i| \\
& y_1 = \sum_{j=1}^{n} a_{1j}x_j - b_1, \\
& y_2 = \sum_{j=1}^{n} a_{2j}x_j - b_2, \\
& \vdots \\
& y_H = \sum_{j=1}^{n} a_{Hj}x_j - b_H.
\end{array}
$$

$$
\begin{cases}
y_i \geq \sum_{j=1}^{n} a_{ij}x_j - b_i \\
y_i \geq -(\sum_{j=1}^{n} a_{ij}x_j - b_i)
\end{cases}
$$

Equivalent LP formulation

$$
\begin{array}{lll}
\min & \sum_{k=1}^{H}(y'_k + y''_k) & \\
& y'_k - y''_k = \sum_{j=1}^{n} a_{kj}x_j - b_k & \forall k = 1,2\dots,H
\end{array}
$$

ISE 426
Optimization models and applications

Lecture 14 — October 20, 2015

- Goal programming
- Winston & Venkataramanan, pages 191-194

Goal programming

Consider the following management problem[1]:

- A company is introducing three new products, P1, P2, P3, and wants to know how many such products to make.
- The per-unit profit, per-unit employment level, and the per-unit capital investment for each product are as follows:

	P1	P2	P3
profit [M$]	12	9	15
employment lev. [×100 wrks.]	5	3	4
capital inv. [M$]	4	7	8

The company:

- wants to have at least 125 M$ profit
- wants to keep its 4,000 employees, no more, no less
- wants to invest less than 55 M$

[1]Similar example on Winston & Venkataramanan, p. 191.

Optimization model

Variables: x_1, x_2, x_3.

Constraints:

- Profit: $12x_1 + 9x_2 + 15x_3 \geq 125$
- Employees: $5x_1 + 3x_2 + 4x_3 = 40$
- Investment: $5x_1 + 7x_2 + 8x_3 \leq 55$

Now we'd have to find an objective function.

However, a quick check with a dummy objective (e.g. x_1) tells us that the problem is infeasible: there is no value of (x_1, x_2, x_3) that satisfies **all** constraints.

Now who tells them the problem is infeasible?

The company now says that some constraints are **not** strict: they may be violated, but not too much.

They are goals: instead of focusing on one single objective function, try to make as many as possible to be satisfied as much as posisble.

Previous estimations give the per-unit loss associated with violation of each constraint.

- 5 M$ per-unit loss for the long-run profit constraint (<125)

i.e. if we find a solution with $12x_1 + 9x_2 + 15x_3 = 122$, we'll incur losses for $5 \times (125 - 122) = 15$ M$

- 4 M$ per-unit loss when number of employees <40
- 2 M$ per-unit loss when number of employees >40
- 3 M$ per-unit loss when capital investment >55

Modify model: non-preemptive Goal Programming

- One or more constraint needs to be relaxed.
- Instead of ignoring them, penalize their **violation**:

$$\begin{array}{rrrcrrr} 12x_1 & +9x_2 & +15x_3 & \geq & 125 & & -y_1^- \\ 5x_1 & +3x_2 & +4x_3 & = & 40 & +y_2^+ & -y_2^- \\ 5x_1 & +7x_2 & +8x_3 & \leq & 55 & +y_3^+ & \end{array}$$

with $y_1^-, y_2^+, y_2^-, y_3^+ \geq 0$

- We'd like y_1^-, y_2^+, y_2^-, and y_3^+ to be all zero, but this is not possible as the problem would be infeasible.

$\Rightarrow$ try to make them as small as possible

Non-preemptive goal programming assumes all goals should be pursued (each with a weight).

minimize the deviation.

Non-preemptive Goal Programming

duality slackness.

$$\min \left(5y_1^- \; +2y_2^+ \; +4y_2^- \; +3y_3^+ \right)$$

$$\begin{array}{rrrcrrr} 12x_1 & +9x_2 & +15x_3 & \geq & 125 & & -y_1^- \\ 5x_1 & +3x_2 & +4x_3 & = & 40 & +y_2^+ & -y_2^- \\ 5x_1 & +7x_2 & +8x_3 & \leq & 55 & +y_3^+ & \end{array}$$

$y_1^-, y_2^+, y_2^-, y_3^+ \geq 0$

- Result: $(x_1, x_2, x_3) = (\frac{25}{3}, 0, \frac{5}{3})$, and the only constraint being really relaxed is the second: $5x_1 + 3x_2 + 4x_3 = \frac{145}{3} = 48.333 > 40$.

i.e. $(y_1^-, y_2^+, y_2^-, y_3^+) = (0, 8.333, 0, 0)$

- Now at the company they start to think that maybe the second constraint **is** more important...

Preemptive Goal Programming

We still cannot satisfy all constraints, but we do prefer satisfying some rather than others.

Preemptive goal programming assumes some goals are more important than others, and satisfying the former should be a priority over the latter.

For the company, the main priorities are

- to preserve the total capital, and
- to keep employment level at most 40 (only one half of the second constraint), i.e. don't want to hire!

Once these are respected, we also care about the remaining two constraints:

- to do at least 125 M$ profit
- to keep employment level at least 40, i.e. don't want to fire

How do we model this?

Preemptive Goal Programming

For each **goal**, from most important to least important:

1. ignore (=relax) constraints at all lower levels
2. add penalization terms for this goal to objective function
3. solve
4. fix maximum violation of priorities at current level

Preemptive Goal Programming: stage 1

$$
\begin{array}{llllllll}
\min & 2y_2^+ & +3y_3^+ & & & & & \\
& 12x_1 & +9x_2 & +15x_3 & \geq & 125 & & -y_1^- \\
& 5x_1 & +3x_2 & +4x_3 & = & 40 & +y_2^+ & -y_2^- \\
& 5x_1 & +7x_2 & +8x_3 & \leq & 55 & +y_3^+ & \\
& & y_1^-, y_2^-, & y_2^+, y_3^+ & \geq & 0 & &
\end{array}
$$

- Only the violations of the more important constraints (y_2^+ and y_3^+) appear in the objective

$\Rightarrow$ The others don't, their constraints are **ignored** (=relaxed)

Result: $(x_1, x_2, x_3) = (0, 0, 0)$ (oops...), but we managed to satisfy both "important" constraints ($y_2^+ = y_3^+ = 0$).

The first constraint was relaxed, so it's easy to select x_i such that the other two are satisfied.

infinite not only (0,0,0).

Preemptive Goal Programming: stage 2

$$
\begin{array}{llllllll}
\min & 5y_1^- & +2y_2^+ & +4y_2^- & +3y_3^+ & & & \\
& 12x_1 & +9x_2 & +15x_3 & \geq & 125 & & -y_1^- \\
& 5x_1 & +3x_2 & +4x_3 & = & 40 & +y_2^+ & -y_2^- \\
& 5x_1 & +7x_2 & +8x_3 & \leq & 55 & +y_3^+ & \\
& & & y_2^+ & = & 0 & & \\
& & & y_3^+ & = & 0 & & \\
& & & y_1^-, y_2^- & \geq & 0 & &
\end{array}
$$

- violation is fixed to 0 for the important constraints
- violation of the secondary constraints appear in the objective

Result: $(x_1, x_2, x_3) = (5, 0, 3.75)$, only the first constraint is violated (profit is $125 - y_1^- = 125 - 8.75 = 116.25$).

Example

The city council is developing an equitable city rate tax table. Taxes come from a combination of four sources:

- Property taxes: ($550M base)
- Food & Drugs: ($35M base)
- Other Sales: ($55M base)
- Gasoline: (Consumption: 7.5 million gallons/year)

They would like to come up with a "fair" city tax...

- Tax revenues must be at least $16M
- The property tax rate should be $\leq 1\%$.
- Food/drug taxes must be $\leq 10\%$ of all taxes collected
- Sales taxes must be $\leq 20\%$ of all taxes collected
- The gasoline tax must be $\leq \$0.02$/gallon.

Model

Variables:

r_p	Property tax rate	x_p	Property tax collected (in $)
r_f	Food/Drug tax rate	x_f	Food/Drug collected (in $)
r_s	Sales tax rate	x_s	Sales tax collected (in $)
r_g	Gas tax rate [$/gallon]	x_g	Gas tax collected (in $)
		T	Total taxes collected (in $)

Taxes (from rates) are based on the tax base

Model

Constraints (definition of x variables):

$$\begin{aligned} x_p &= 550r_p \\ x_f &= 35r_f \\ x_s &= 55r_s \\ x_g &= 7.5r_g \\ T &= x_p + x_f + x_s + x_g \end{aligned}$$

Requirements Constraints:

Revenue:	$T \geq 16$
Property Tax Rate:	$r_p \leq 0.01$
Food-Drug tax restriction:	$x_f \leq 0.1T$
Sales tax restriction:	$x_s \leq 0.2T$
Gas tax restriction:	$r_g \leq 0.02$

What's the objective?
It doesn't matter: The problem is infeasible!

Non-preemptive goal programming

Minimize the sum of violations altogether

$$\begin{aligned} \min \quad & e_p + e_f + e_s + e_g \\ & x_p = 550r_p \\ & x_f = 35r_f \\ & x_s = 55r_s \\ & x_g = 7.5r_g \\ & T = x_p + x_f + x_s + x_g \\ & T \geq 16 \\ & r_p \leq 0.01 \quad +e_p \\ & x_f \leq 0.1T \quad +e_f \\ & x_s \leq 0.2T \quad +e_s \\ & r_g \leq 0.02 \quad +e_g \\ & e_p, e_f, e_s, e_g \geq 0 \end{aligned}$$

Preemptive goal programming

Minimize each violation separately, in order. Suppose the order is (e_p, e_f, e_s, e_g). **Step 1**:

$$\begin{aligned} \min \quad & e_p \\ & x_p = 550r_p \\ & x_f = 35r_f \\ & x_s = 55r_s \\ & x_g = 7.5r_g \\ & T = x_p + x_f + x_s + x_g \\ & T \geq 16 \\ & r_p \leq 0.01 \quad +e_p \\ & x_f \leq 0.1T \quad +e_f \\ & x_s \leq 0.2T \quad +e_s \\ & r_g \leq 0.02 \quad +e_g \\ & e_p, e_f, e_s, e_g \geq 0 \end{aligned}$$

$\Rightarrow$ Result: $e_p = 0$ many solutions to get

Preemptive goal programming: step 2

$$\begin{aligned} \min \quad & e_f \\ & x_p = 550r_p \\ & x_f = 35r_f \\ & x_s = 55r_s \\ & x_g = 7.5r_g \\ & T = x_p + x_f + x_s + x_g \\ & T \geq 16 \\ & r_p \leq 0.01 \quad +e_p \\ & x_f \leq 0.1T \quad +e_f \\ & x_s \leq 0.2T \quad +e_s \\ & r_g \leq 0.02 \quad +e_g \\ & e_p = 0, e_f, e_s, e_g \geq 0 \end{aligned}$$

$\Rightarrow$ Result: $e_f = 0$

Preemptive goal programming: step 3

$$\begin{aligned} \min \quad & e_s \\ & x_p = 550r_p \\ & x_f = 35r_f \\ & x_s = 55r_s \\ & x_g = 7.5r_g \\ & T = x_p + x_f + x_s + x_g \\ & T \geq 16 \\ & r_p \leq 0.01 \quad +e_p \\ & x_f \leq 0.1T \quad +e_f \\ & x_s \leq 0.2T \quad +e_s \\ & r_g \leq 0.02 \quad +e_g \\ & e_p = 0, e_f = 0, e_s, e_g \geq 0 \end{aligned}$$

$\Rightarrow$ Result: $e_s = 0$

Preemptive goal programming: step 4

$$\begin{aligned} \min \quad & e_g \\ & x_p = 550r_p \\ & x_f = 35r_f \\ & x_s = 55r_s \\ & x_g = 7.5r_g \\ & T = x_p + x_f + x_s + x_g \\ & T \geq 16 \\ & r_p \leq 0.01 \quad +e_p \\ & x_f \leq 0.1T \quad +e_f \\ & x_s \leq 0.2T \quad +e_s \\ & r_g \leq 0.02 \quad +e_g \\ & e_p = 0, e_f = 0, e_s = 0, e_g \geq 0 \end{aligned}$$

$\Rightarrow$ Result: $e_g = 0.74$

Min-max goal programming

Minimize the maximum violation

$$
\begin{aligned}
\min \quad & \max\{e_p, e_f, e_s, e_g\} \\
& x_p = 550 r_p \\
& x_f = 35 r_f \\
& x_s = 55 r_s \\
& x_g = 7.5 r_g \\
& T = x_p + x_f + x_s + x_g \\
& \\
& T \geq 16 \\
& r_p \leq 0.01 \quad + e_p \\
& x_f \leq 0.1T \quad + e_f \\
& x_s \leq 0.2T \quad + e_s \\
& r_g \leq 0.02 \quad + e_g \\
& e_p, e_f, e_s, e_g \geq 0
\end{aligned}
$$

Min-max goal programming

$$
\begin{aligned}
\min \quad & z \\
& z \geq e_p \\
& z \geq e_f \\
& z \geq e_s \\
& z \geq e_g \\
& x_p = 550 r_p \\
& x_f = 35 r_f \\
& x_s = 55 r_s \\
& x_g = 7.5 r_g \\
& T = x_p + x_f + x_s + x_g \\
& T \geq 16 \\
& r_p \leq 0.01 \quad + e_p \\
& x_f \leq 0.1T \quad + e_f \\
& x_s \leq 0.2T \quad + e_s \\
& r_g \leq 0.02 \quad + e_g \\
& e_p, e_f, e_s, e_g \geq 0
\end{aligned}
$$

Result: $e_p = e_f = e_s = e_g = 0.00991957$

ISE 426
Optimization models and applications

Lecture 15 — October 22, 2015

- Integer (Linear) Programming (IP)
- Examples

Reading:

- Winston & Venkataramanan, Chapter 9

Mixed-Integer Linear Programming (MILP) problems

…or more simply, Integer Programming (IP) problems:

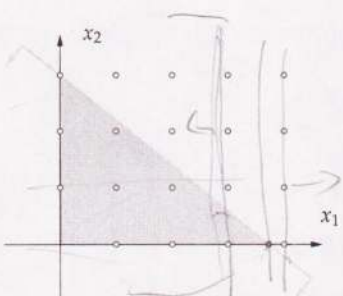
$$\begin{array}{llllll}
\min & c_1x_1 & +c_2x_2 & \dots & +c_nx_n & \\
& a_{11}x_1 & +a_{12}x_2 & \dots & +a_{1n}x_n & \le b_1 \\
& a_{21}x_1 & +a_{22}x_2 & \dots & +a_{2n}x_n & \le b_2 \\
& \vdots & & & & \\
& a_{m1}x_1 & +a_{m2}x_2 & \dots & +a_{mn}x_n & \le b_m \\
& & & & x_i & \in \mathbb{Z} \quad \forall i \in J \subseteq \{1,2\dots,n\}
\end{array}$$

A much more powerful modeling tool than LP:

yes/no decisions ($x_i \in \{0,1\}$)	nonconvexities
integer quantities	discontinuities
piece-wise linear functions	economies of scale

Much more difficult than LP models: nonconvex

Why can't we just round numbers up/down?



$$\begin{array}{lll}
\max & x_1 & +1.1x_2 \\
& \frac{x_1}{3.7} & +\frac{x_2}{3} \le 1 \\
& & x_1, x_2 \in \mathbb{Z}_+
\end{array}$$

- Optimal solution of the LP relaxation: $(3.7, 0)$, obj. f.: 3.7
- Rounded solution: $(3, 0)$, obj. f.:3.0
- Optimal solution of the original problem: $(0, 3)$, obj. f.: 3.3

relaxation + B&B
may produce gap. 0.3.
10% not big

LP and IP

IP is a bit younger than LP, but is the subject of extensive research and can model countless problems in Industry:

- Airline crew scheduling
- Vehicle Routing
- Financial applications
- Design of Telecommunications networks

IP problems are difficult to solve:

- Require very specialized techniques
  (some use LP relaxations to find lower bounds)
- A good model makes a problem easier (but not easy)

i.e. unlike LP, the way we model an Optimization problem **affects** the chances to solve it

Binary variables, logical operators

- model yes/no decisions: $x_i \in \{0,1\}$
- $x_i = 0$ if the decision is "no",
- $x_i = 1$ if it is "yes"
- can use logical operators: implications, disjunctions, etc.:
  - Mario or Luigi will have ice cream, but not both: $x_{\text{Mario}} + x_{\text{Luigi}} \le 1$
  - At least one among Mario and Luigi will have ice cream: $x_{\text{Mario}} + x_{\text{Luigi}} \ge 1$
  - If Mario has ice cream, then Giovanni will have one too: $x_{\text{Mario}} \le x_{\text{Giovanni}}$
  - Luigi gets ice cream if and only if Paolo does not get any: $x_{\text{Luigi}} = 1 - x_{\text{Paolo}}$

Binary variables and operations with sets

Binary variables are useful to model problems on sets. E.g.:

- Choose a subset S of a set A of elements such that S has certain properties (e.g. not more than K elements, etc.)
- Each element $i \in A$ has a cost c_a

$\Rightarrow$ The cost of a solution S is $\sum_{i \in S} c_i$

- Define variable x_i:

$$x_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{otherwise} \end{cases}$$

- now the cost of a solution S is $\sum_{i \in A: x_i = 1} c_i = \sum_{i \in A} c_i x_i$
- define properties similarly, e.g. $|S| \le K$ is $\sum_{i \in A} x_i \le K$

Example: Subset Sum

Two brothers, Ludwig and Johann, inherit from their dear uncle a set A of antique objects.

- each worth a lot of money, c_i for all $i \in A$
- they want to share these objects in a balanced manner

$\Rightarrow$ minimize the difference between their total values

- $S_L \subset A$ contains the objects that Ludwig will get, while $S_J = A \setminus S_L$ are the remaining objects

$\Rightarrow$ minimize

$$\left| \sum_{i \in S_L} c_i - \sum_{i \in S_J} c_i \right|$$

How to model this with IP?

Example: Subset Sum (cont'd)

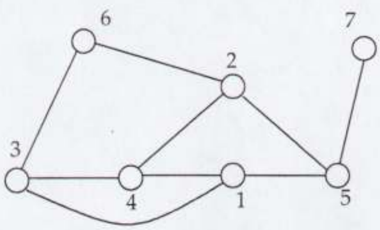
$$x_i = \begin{cases} 1 & \text{if Ludwig gets the } i\text{-th object} \\ 0 & \text{if Johann gets the } i\text{-th object} \end{cases}$$

Then the integer model is

$$\begin{aligned} \min \quad & \left| \textstyle\sum_{i \in A} c_i x_i - \sum_{i \in A} c_i (1 - x_i) \right| \\ & x_i \in \{0, 1\} \qquad \forall i \in A \end{aligned}$$

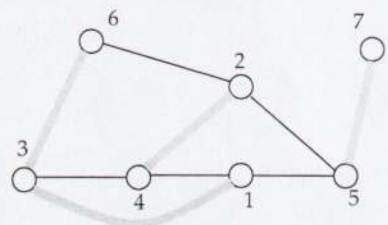
Example: the edge covering problem

In a graph $G = (V, E)$ as in the figure, choose a subset S of edges such that all nodes are "covered" by at least one edge in S. Minimize the number of edges used



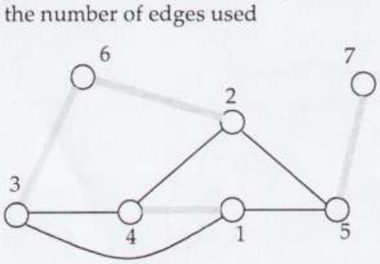
Example: the edge covering problem

In a graph $G = (V, E)$ as in the figure, choose a subset S of edges such that all nodes are "covered" by at least one edge in S. Minimize the number of edges used



Example: the edge covering problem

In a graph $G = (V, E)$ as in the figure, choose a subset S of edges such that all nodes are "covered" by at least one edge in S. Minimize the number of edges used



Edge covering

$$\begin{array}{lcccccccccl}
\min & x_{13} & +x_{14} & +x_{15} & +x_{24} & +x_{25} & +x_{26} & +x_{34} & +x_{36} & +x_{57} & \\
& x_{13} & +x_{14} & +x_{15} & & & & & & & \geq 1 \\
& & & & x_{24} & +x_{25} & +x_{26} & & & & \geq 1 \\
& x_{13} & & & & & & +x_{34} & +x_{36} & & \geq 1 \\
& & x_{14} & & +x_{24} & & & +x_{34} & & & \geq 1 \\
& & & x_{15} & & +x_{25} & & & & +x_{57} & \geq 1 \\
& & & & & & +x_{26} & & +x_{36} & & \geq 1 \\
& & & & & & & & & x_{57} & \geq 1 \\
& x_{13}, & x_{14}, & x_{15}, & x_{24}, & x_{25}, & x_{26}, & x_{34}, & x_{36}, & x_{57} & \in \{0, 1\}
\end{array}$$

Edge covering

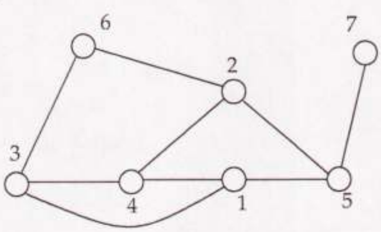
$$
\begin{array}{lll}
\min & \sum_{\{i,j\}\in E} x_{\{i,j\}} & \\
& \sum_{j\in V:\{i,j\}\in E} x_{\{i,j\}} \geq 1 & \forall i \in V \\
& x_{\{i,j\}} \in \{0,1\} & \forall \{i,j\} \in E
\end{array}
$$

Graphs and AMPL

```
param n; # number of nodes
set   V=1..n; # set of nodes
set   E within {i in V, j in V: i<j};
  # subset of set of node pairs

var x {E} binary;

minimize numEdges: sum {(i,j) in E} x [i,j];

covering {i in V}:
  sum {j in V: (i,j) in E} x[i,j] +
  sum {j in V: (j,i) in E} x[j,i] >= 1;

data;

param n := 7;
set E := (1,3) (1,4) (1,5) (2,4) (2,5)
         (2,6) (3,4) (3,6) (5,7);
```

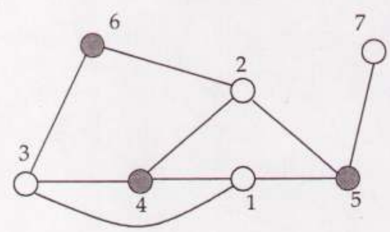
Example: the node packing (or stable set) problem

In a graph $G = (V, E)$ as in the figure, choose a subset S of nodes such that no two nodes i and j in S are adjacent, i.e. share an edge $\{i, j\}$. In other words, if both i and j are included in S, then there must be no edge $\{i, j\}$. Maximize the number of nodes used.



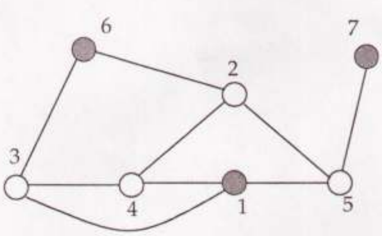
Example: the node packing (or stable set) problem

In a graph $G = (V, E)$ as in the figure, choose a subset S of nodes such that no two nodes i and j in S are adjacent, i.e. share an edge $\{i, j\}$. In other words, if both i and j are included in S, then there must be no edge $\{i, j\}$. Maximize the number of nodes used.



Example: the node packing (or stable set) problem

In a graph $G = (V, E)$ as in the figure, choose a subset S of nodes such that no two nodes i and j in S are adjacent, i.e. share an edge $\{i, j\}$. In other words, if both i and j are included in S, then there must be no edge $\{i, j\}$. Maximize the number of nodes used.



Node packing

$$
\begin{array}{lcccccccl}
\max & x_1 & +x_2 & +x_3 & +x_4 & +x_5 & +x_6 & +x_7 & \\
& x_1 & & +x_3 & & & & & \leq 1 \\
& x_1 & & & +x_4 & & & & \leq 1 \\
& x_1 & & & & +x_5 & & & \leq 1 \\
& & x_2 & & +x_4 & & & & \leq 1 \\
& & x_2 & & & +x_5 & & & \leq 1 \\
& & x_2 & & & & +x_6 & & \leq 1 \\
& & & x_3 & +x_4 & & & & \leq 1 \\
& & & x_3 & & & +x_6 & & \leq 1 \\
& & & & & x_5 & & +x_7 & \leq 1
\end{array}
$$

Node packing

$$
\begin{array}{lll}
\max & \sum_{i \in V} x_i & \\
& x_i + x_j \leq 1 & \forall \{i,j\} \in E \\
& x_i \in \{0,1\} & \forall i \in V
\end{array}
$$

Node packing in AMPL

```
param n; # number of nodes
set   V=1..n; # set of nodes
set   E within {i in V, j in V: i<j};
  # subset of set of node pairs

var x {V} binary;

maximize numNodes: sum {i in V} x [i];

packing {(i,j) in E}: x [i] + x [j] <= 1;

data;

param n := 7;
set E := (1,3) (1,4) (1,5) (2,4) (2,5)
         (2,6) (3,4) (3,6) (5,7);
```

Binary variables and *fixed charge* quantities

Binary variables are also useful to model economies of scale:

- fixed transaction costs in financial optimization
- fixed costs for using a facility/plant/service

Example: "Every truck in the fleet costs \$25,000/year for maintenance plus \$1.26 per mile. Determine the trucks to be used and the distance they will travel."

$\Rightarrow$ Use a binary variable $x_i = \begin{cases} 1 & \text{if truck } i \text{ is used} \\ 0 & \text{otherwise} \end{cases}$

... and a continuous variable y_i = miles traveled by truck i

The objective function will look like $\$25,000 x_i + \$1.26 y_i$.
(Implicit) constraint: if truck i travels half a mile, it has to be bought: $y_i > 0 \Rightarrow x_i = 1$, or

$$y_i \leq M x_i$$

where M is an adequate constant.

Example: Production planning with fixed costs

A small firm produces plastic for the car industry.

- At the beginning of the year, it knows exactly the demand d_i of plastic for every month i.
- It also has a maximum production capacity of P and an inventory capacity of C.
- The inventory is empty on 01/01 and has to be empty again on 12/31
- production has a monthly per-unit cost c_i
- \+ a monthly **fixed** cost f_i if the machinery for producing plastic is started in month i

What do we produce at each month to minimize total production cost while satisfying demand?

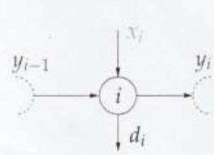
Production planning with fixed costs

- How much to produce each month i: x_i, $i = 1, 2 \ldots, 12$
- The inventory level at the end of each month (demand is satisfied by part of production **and** part of inventory): y_i, $i = 0, 1, 2 \ldots, 12$
- Why 0? Because we need to constrain the inventory on 01/01 (as well as on 12/31)
- \+ a binary variable $z_i = \begin{cases} 1 & \text{if machine started in month } i \\ 0 & \text{otherwise} \end{cases}$

Production planning. What constraints?

- Production capacity constraint: $x_i \leq P \quad \forall i = 1, 2 \ldots, 12$
- Beginning/end of year: $y_0 = 0, \quad y_{12} = 0$
- What goes in must go out...
- \+ production in month i only occurs[1] when the machine is used in i, i.e. $x_i > 0 \Rightarrow z_i = 1$:

$$x_i \leq P z_i$$



$\Rightarrow \quad x_i + y_{i-1} = d_i + y_i \quad \forall i = 1, 2 \ldots, 12$

[1]That is, it can be nonzero and at most P

Production planning model

$$\min \sum_{i=1}^{12}(c_i x_i + f_i z_i)$$
$$x_i + y_{i-1} = d_i + y_i \quad \forall i = 1, 2 \ldots, 12$$
$$0 \le x_i \le P z_i \quad \forall i = 1, 2 \ldots, 12$$
$$y_i \ge 0, z_i \in \{0, 1\} \quad \forall i = 1, 2 \ldots, 12$$
$$y_0 = y_{12} = 0$$

Note: $x_i \le P z_i$ implies $x_i \le P$ as $z_i \le 1$ $y_i \le C$ $\forall i = 1, 2, \ldots$

Production planning model

```
set Months     = 1..12;
set MonthsPlus = 0..12;

param cost       {Months};
param fixed_cost {Months};
param demand     {Months};
param Capacity;

var production {Months}     >= 0;
var inventory  {MonthsPlus} >= 0;
var use        {Months} binary;

minimize prodCost:
  sum {i in Months} (cost       [i] * production [i] +
                     fixed_cost [i] * use        [i]);
```

Production planning model

```
conservation {i in Months}:
  production [i] + inventory [i-1] =
  demand     [i] + inventory [i];

startMachine {i in Months}:
  production [i] <= Capacity * use [i];

Jan1Inv:  inventory  [0] = 0;
Dec31Inv: inventory [12] = 0;
```

Problem data

```
param Capacity = 120;

param cost :=
1 14      2 19      3 15      4 11
5 10      6  7      7  4      8  5
9  7     10 10     11 11     12 13;

param demand :=
1 110     2  70     3  85     4 90
5 140     6  90     7  40     8 80
9 100    10 105    11 140    12 80;

param fixed_cost :=
1 435     2  470     3  425     4 390
5 340     6  290     7  240     8 280
9 300    10  345    11  390    12 380;
```

An "adequate" constant???

- Constraints such as $y_i \le M x_i$ are formally correct, but they are **horrible** for an MILP solver. The bigger M, the uglier.
- We still must ensure that y_i can take any feasible value.

⇒ If you know an explicit upper bound on y_i, use it.

- Physical constraint: for a truck traveling at 65mph, 24 hours a day, 365 days, $M = 569,400$mi (very ugly).
- "The company only works on weekdays 9am-5pm, drivers have an hour lunch break. Loading and unloading a truck takes 30 minutes"

⇒ 65mph × (8h − 1h − 2 × 0.5h)/day × 200days/yr = 78,000mi

Binary Linear problem

BLP

relaxation → bound.

If M is much big.
in relaxation, x_i will be very small.
in objective $\min \sum c_i x_i$.
like: feasibe: $x_i = 1$
relaxation: $x_i = 10^{-n}$.
there is a big gap !!

ISE 426
Optimization models and applications

Lecture 16 — October 27, 2015

- More IP
- $a^\top x \le b \;\Rightarrow\; c^\top x \le d$
- Extreme points and relaxations

Fun with logic

Consider propositions $a, b, c \ldots$, all in $\{T, F\}$. They can all be modeled with binary variables $x_a, x_b, x_c \ldots \in \{0, 1\}$
The negation of a is denoted as $\neg a$ or $\bar{a}$.

- $a \vee b$ (i.e., "$a \vee b$ is true") becomes $x_a + x_b \ge 1$
- $a \wedge b$ becomes $x_a = 1, x_b = 1$ or $x_a + x_b = 2$ (trivial!)
- $\neg a$ becomes $x_a = 0$

  Examples:
- $\neg(a \vee b)$ becomes $\neg(x_a + x_b \ge 1)$ or $x_a + x_b = 0$ or $x_a = x_b = 0$
- $a \vee \neg b$ becomes $x_a + (1 - x_b) \ge 1$ or $x_a \ge x_b$
- $\neg(a \wedge b)$ becomes $x_a + x_b \le 1$
- $a \wedge \neg b$ becomes $x_a = 1, x_b = 0$ or $x_a + (1 - x_b) = 2$

More fun with logic

- $a \Rightarrow b$ becomes $x_a \le x_b$
- $a \Leftrightarrow b$ becomes $x_a = x_b \quad (x_a \le x_b, \quad x_b \le x_a)$

- $c \Rightarrow a \vee b$ becomes $x_c \le x_a + x_b$
- $c \Leftarrow a \vee b$ becomes $x_c \ge x_a, x_c \ge x_b$
- $c \Leftrightarrow a \vee b$ becomes $x_c \ge x_a, x_c \ge x_b, x_c \le x_a + x_b$

- $c \Rightarrow a \wedge b$ becomes $x_c \le x_a, x_c \le x_b$
- $c \Leftarrow a \wedge b$ becomes $x_c \ge x_a + x_b - 1$
- $c \Leftrightarrow a \wedge b$ becomes $x_c \le x_a, x_c \le x_b, x_c \ge x_a + x_b - 1$

Examples

C_2: $\Leftrightarrow (x_2 = 1)$ C_3: $\Leftrightarrow (x_3 = 1)$.

"$(a \wedge \neg b \wedge \neg c) \vee (b \wedge \neg c) \vee (\neg a \wedge c)$ is true" becomes

C_1: $\Leftrightarrow (x_1 = 1)$.

$$
\begin{aligned}
& x_1 + x_2 + x_3 \ge 1 \\
& x_1 \ge x_a + (1 - x_b) + (1 - x_c) - 2 \\
& x_1 \le x_a \\
& x_1 \le 1 - x_b \\
& x_1 \le 1 - x_c \\
& x_2 \ge x_b + (1 - x_c) - 1 \\
& x_2 \le x_b \\
& x_2 \le 1 - x_c \\
& x_3 \ge (1 - x_a) + x_c - 1 \\
& x_3 \le 1 - x_a \\
& x_3 \le x_c
\end{aligned}
$$

Examples

"$(a \wedge b \wedge \neg c) \vee (\neg a \wedge b \wedge \neg c) \Rightarrow (\neg a \wedge d \wedge e)$ is true" becomes

$$
\begin{aligned}
& x_1 \le x_3 \\
& x_2 \le x_3 \\
& x_1 \ge x_a + x_b + (1 - x_c) - 2 \\
& x_1 \le x_a \\
& x_1 \le x_b \\
& x_1 \le 1 - x_c \\
& x_2 \ge (1 - x_a) + x_b + (1 - x_c) - 2 \\
& x_2 \le 1 - x_a \\
& x_2 \le x_b \\
& x_2 \le 1 - x_c \\
& x_3 \ge (1 - x_a) + x_d + x_e - 2 \\
& x_3 \le 1 - x_a \\
& x_3 \le x_d \\
& x_3 \le x_e
\end{aligned}
$$

Switching constraints on/off

Suppose constraint $a_1x_1 + a_2x_2 + \ldots + a_nx_n \le b$, or $a^\top x \le b$ for short, depends on the value a binary variable y.

- for instance, it is implied by $y = 1$
- or it implies $y = 1$ when satisfied

How do we model this in MILP? The constraint can still be linear, but it will depend on y.

Switching constraints on/off

Consider $y = 1 \Rightarrow a^\top x \le b$.

- constraint $a^\top x \le b$ is mandatory if $y = 1$.
- it is not if $y = 0$, that is, if $y = 0$ then $a^\top x \le +\infty$

We need to unify these two constraints. Easy! We only need to change the right-hand side (rhs) of $a^\top x \le b$:

$$a^\top x \le b + M(1-y) \quad \Leftrightarrow \quad \begin{cases} a^\top x \le b + M \approx +\infty & \text{if } y = 0 \\ a^\top x \le b & \text{if } y = 1 \end{cases}$$

$y = 1 \Rightarrow a^\top x \ge b$ translates to $a^\top x \ge b - M'(1-y)$.

What about $y = 1 \Rightarrow a^\top x = b$? It's equivalent to $y = 1 \Rightarrow (a^\top x \ge b, a^\top x \le b)$:

$$\begin{cases} a^\top x \le b + M(1-y) \\ a^\top x \ge b - M'(1-y) \end{cases}$$

M and M' are positive and, in general, different.

Switching constraints on/off

Now the equivalence: $y = 1 \Leftrightarrow a^\top x \le b$. It means

$$\begin{array}{ll} y = 1 \Rightarrow a^\top x \le b & \text{(and)} \\ a^\top x \le b \Rightarrow y = 1 & \end{array}$$

or equivalently

$$\begin{array}{ll} y = 1 \Rightarrow a^\top x \le b & \text{(and)} \\ y = 0 \Rightarrow a^\top x > b & \end{array}$$

We can model the first as $a^\top x \le b + M(1-y)$; the second is:

$$a^\top x > b - M'y$$

In general, the ">" or "<" are not accepted, as it depends on the precision of solver/modeler/computer (e.g. 10^{-15}). Same for "$\neq$". Use small numbers:

$$a^\top x \ge b + \epsilon - M'y$$

What are good values of M and M'?

$$a^\top x \le b + M(1-y) \qquad a^\top x \ge b - M'(1-y)$$

- Suppose each variable x_i has lower/upper bounds $l_i \le x_i \le u_i$. For short, let's use $l \le x \le u$ or $x \in [l, u]$
- Otherwise, use huge bounds $-10^{20} \le x_i \le 10^{20}$ (bad!)
- $a^\top x \le +\infty$ means "$a^\top x$ is at most the maximum value it can take with $x \in [l, u]$" (redundant: $a^\top x$ can be anything)
- $a^\top x \ge -\infty$ means "$a^\top x$ is at least the minimum value it can take with $x \in [l, u]$" (likewise)
- If $a \ge 0$, i.e. if all $a_i \ge 0$, then $M = au$ and $M' = al$

$\Rightarrow$ M is $\sum_{i:a_i>0} a_i u_i + \sum_{i:a_i<0} a_i l_i - b$

$\Rightarrow$ M' is $b - (\sum_{i:a_i>0} a_i l_i + \sum_{i:a_i<0} a_i u_i)$

Example 1

$$\begin{array}{rcl} y = 1 \Rightarrow 3x_1 + 5x_2 - 2x_3 \le 6 \\ -7 \le & x_1 & \le 4 \\ 0 \le & x_2 & \le 12 \\ -11 \le & x_3 & \le -1 \\ & y \in \{0, 1\} & \end{array}$$

becomes $3x_1 + 5x_2 - 2x_3 \le 6 + M(1-y)$, with

$$M = 3 \cdot 4 + 5 \cdot 12 - 2 \cdot (-11) - 6 = 94 - 6 = 88$$

which means

$$3x_1 + 5x_2 - 2x_3 \le \begin{cases} 6 & \text{if } y = 1 \\ 6 + 88 = 94 & \text{if } y = 0 \end{cases}$$

94 maximizes $3x_1 + 5x_2 - 2x_3$ for x_1, x_2, x_3 in their bounds.

It is a redundant rhs when $y = 0$ (exactly what we need).

Example 2

$$\begin{array}{rcl} y = 1 \Rightarrow 3x_1 + 5x_2 - 2x_3 \ge 6 \\ -7 \le & x_1 & \le 4 \\ 0 \le & x_2 & \le 12 \\ -11 \le & x_3 & \le -1 \\ & y \in \{0, 1\} & \end{array}$$

becomes $3x_1 + 5x_2 - 2x_3 \ge 6 - M'(1-y)$, with

$$M' = 6 - (3 \cdot -7 + 5 \cdot 0 - 2 \cdot (-1)) = 6 - (-19) = 25$$

which means

$$3x_1 + 5x_2 - 2x_3 \ge \begin{cases} 6 & \text{if } y = 1 \\ 6 - 25 = -19 & \text{if } y = 0 \end{cases}$$

-19 minimizes $3x_1 + 5x_2 - 2x_3$ for x_1, x_2, x_3 in their bounds

It is a redundant rhs when $y = 0$ (exactly what we need).

Example 3

$$\begin{array}{rcl} y = 1 \Leftrightarrow 3x_1 + 5x_2 - 2x_3 \le 6 \\ -7 \le & x_1 & \le 4 \\ 0 \le & x_2 & \le 12 \\ -11 \le & x_3 & \le -1 \\ & y \in \{0, 1\} & \end{array}$$

becomes

$$\begin{array}{l} 3x_1 + 5x_2 - 2x_3 \le 6 + M(1-y) \\ 3x_1 + 5x_2 - 2x_3 \ge 6 + \epsilon - M'y \end{array}$$

$$\begin{array}{l} M = 3 \cdot 4 + 5 \cdot 12 - 2 \cdot (-11) - 6 = 94 - 6 = 88 \\ M' = 6 + \epsilon - (3 \cdot -7 + 5 \cdot 0 - 2 \cdot (-1)) = 6 - (-19) = 25 + \epsilon \end{array}$$

which means

$$3x_1 + 5x_2 - 2x_3 \begin{cases} \le 6 & \text{if } y = 1 \\ \ge 6 + \epsilon & \text{if } y = 0 \end{cases}$$

Implications among constraints

$$a^\top x \le b \Rightarrow c^\top x \le d$$

is equivalent to

$$a^\top x \le b \Rightarrow y = 1; \qquad y = 1 \Rightarrow c^\top x \le d.$$

$a^\top x \le b \Leftarrow c^\top x \le d$ is equivalent to $c^\top x \le d \Rightarrow a^\top x \le b$.

$\Leftrightarrow y=1$

$a^\top x \le b \Leftrightarrow c^\top x \le d$ is equivalent to $\begin{cases} c^\top x \le d \Rightarrow a^\top x \le b \\ a^\top x \le b \Rightarrow c^\top x \le d \end{cases}$

Now you can also model things such as

$$(a^\top x \le b) \vee \neg(c^\top x \ge d) \Rightarrow (d^\top x \ge e) \wedge ((f^\top x \le g) \vee \neg(h^\top x \ge p))$$

$\Updownarrow$ $y_1=1$ $\quad\Updownarrow$ $y_2=1$ $\quad\Updownarrow$ $y_3=1$ $\quad\Updownarrow$ $y_4=1$ $\quad\Updownarrow$ $y_5=1$

$\underbrace{y_1 \vee \neg y_2}_{z_1} \Rightarrow y_3 \wedge \underbrace{(y_4 \vee \neg y_5)}_{z_2}$ becomes to binary logic, to control the constraints refering to the logic: use $y=1 \Leftrightarrow a^\top x \le b$

Relaxations and efficiency

Integer programming problems:

$$\begin{array}{lllllll} (IP) & \min & c_1x_1 & +c_2x_2 & \dots & +c_nx_n & \\ & & a_{11}x_1 & +a_{12}x_2 & \dots & +a_{1n}x_n & \le b_1 \\ & & a_{21}x_1 & +a_{22}x_2 & \dots & +a_{2n}x_n & \le b_2 \\ & & \vdots & & & & \\ & & a_{m1}x_1 & +a_{m2}x_2 & \dots & +a_{mn}x_n & \le b_m \\ & & & & & x_i \in \mathbb{Z} & \forall i \in J \subseteq \{1,2\dots,n\} \end{array}$$

or, for short,

$$\begin{array}{lll} (IP) & \min & c^\top x \\ & & Ax \le b \\ & & x_i \in \mathbb{Z} \quad \forall i \in J \subseteq N, \end{array}$$

can be solved using their LP relaxation:

$$\begin{array}{lll} (LP) & \min & c^\top x \\ & & Ax \le b. \end{array}$$

A global optimum z of (LP) is a lower bound for (IP).

Relaxations and efficiency

If an optimal solution $x^\star$ of (LP) is feasible for (IP), i.e., for all $i \in J$ we have $x_i^\star \in \mathbb{Z}$, we're done!

This is **not** the case, usually…

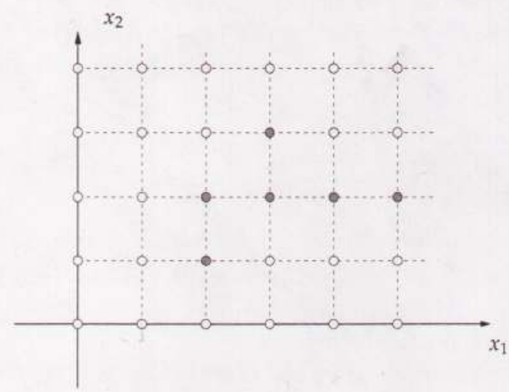
What do we know about the optimal solutions of (LP)? They are all *vertices* of the polyhedron

$$\{x \in \mathbb{R}^n : A^\top x \le b\}$$

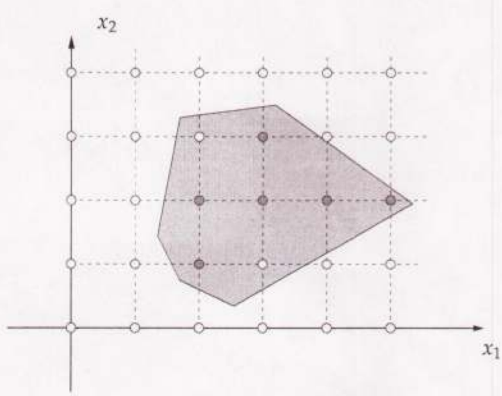
Therefore, it would be just great if all vertices of (LP) were feasible for (IP). Solving IPs would amount to solving LPs, which are a lot easier.

A good model may not achieve just that, but it can help a lot.

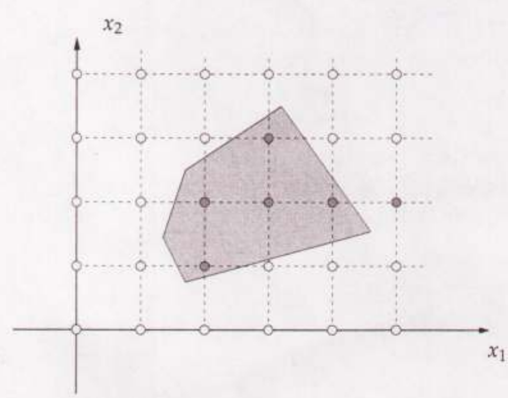
Relaxations, the geometrical standpoint



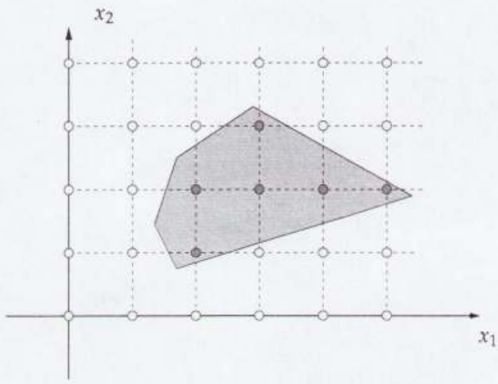
Relaxations, the geometrical standpoint



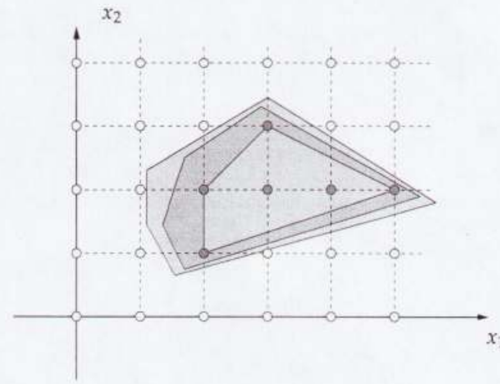
Relaxations, the geometrical standpoint



Relaxations, the geometrical standpoint



Relaxations, the geometrical standpoint



Relaxations: the clique inequality

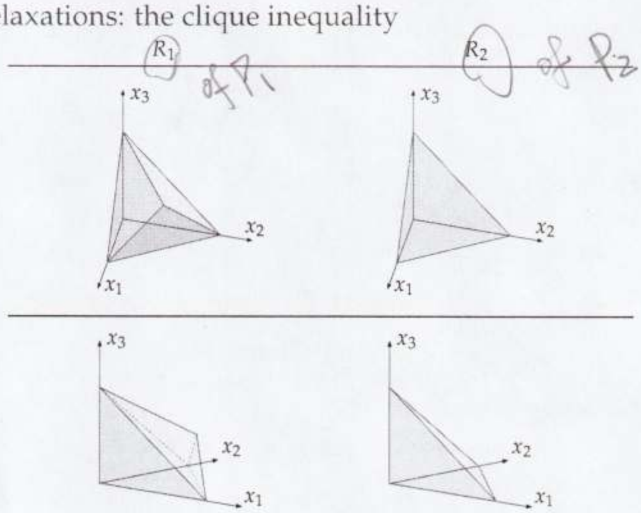
Two models for one problem have the same feasible set and global optima, but may be solved differently:

$$\left.\begin{array}{rl} P_1 : \min & -7x_1 - 8x_2 - 9x_3 \\ \text{s.t.} & x_1 + x_2 \le 1 \\ & x_1 + x_3 \le 1 \\ & x_2 + x_3 \le 1 \\ & x_1, x_2, x_3 \in \{0, 1\} \end{array}\right\} \equiv \left\{\begin{array}{rl} P_2 : \min & -7x_1 - 8x_2 - 9x_3 \\ \text{s.t.} & x_1 + x_2 + x_3 \le 1 \\ & x_1, x_2, x_3 \in \{0, 1\} \end{array}\right.$$

Consider relaxations R_1, R_2 of P_1, P_2 with $x_i \in [0, 1]$.

- R_1: optimal soln. $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, obj.f. -12: lower bound for P_1, P_2
- R_2: optimal solution $(0, 0, 1)$, obj.f. -9: lower bound for P_2 and P_1, and feasible for P_2 and P_1!
- $\Rightarrow$ optimum of P_1, P_2: -9, and P_2 is a better model than P_1

Relaxations: the clique inequality



Good vs. bad models: Uncapacitated Facility Location

A set J of retailers has to be served by a set S of plants, yet to be built. We don't know where the plants will be, but there is a set I of potential sites, and there is

- a cost f_i for building plant $i \in I$
- a (transportation) cost c_{ij} from plant i to retailer j

Each retailer will be served by exactly one plant (why?). Choose a subset S of I such that the total cost is minimized.

Variables:

- $x_i, i \in I$: 1 if plant i is built, 0 otherwise
- y_{ij} assigns retailer j to plant i: 1 if i serves retailer j, 0 otherwise

Good vs. bad models: Uncapacitated Facility Location

Objective function:

$$\sum_{i \in I} f_i x_i + \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij}$$

Constraints:

- for each customer, one facility: $\sum_{i \in I} y_{ij} = 1$
- customers go to mall i if it's there:

$$\sum_{j \in J} y_{ij} \le |J| x_i \quad \forall i \in I$$

or

$$y_{ij} \le x_i \quad \forall i \in I, j \in J$$

Good vs. bad models: Uncapacitated Facility Location

(A)	(B)
$\min \sum_{i\in I} f_i x_i + \sum_{i\in I}\sum_{j\in J} c_{ij} y_{ij}$	$\min \sum_{i\in I} f_i x_i + \sum_{i\in I}\sum_{j\in J} c_{ij} y_{ij}$
$\sum_{i\in I} y_{ij} = 1 \quad \forall i \in I$	$\sum_{i\in I} y_{ij} = 1 \quad \forall i \in I$
$\sum_{j\in J} y_{ij} \le \lvert J\rvert x_i \quad \forall i \in I$	$y_{ij} \le x_i \quad \forall i \in I, j \in J$
$x_i, y_{ij} \in \{0,1\}$	$x_i, y_{ij} \in \{0,1\}$

(A) and (B) are equivalent. However, for $|I| = |J| = 40$,

(A) takes 14 hours[1]

(B) takes 2 seconds

---

[1]On AMPL with an old version of CPLEX.

ISE 426
Optimization models and applications

Lecture 17 — October 29, 2015

- "Good" and "bad" formulations
- Branch&bound for MILP
- Examples of B&B

Reading:

- Hillier & Lieberman, Chapter 13, 13.4 to 13.5
- Winston & Venkataramanan, Chapter 9
- Winston, Chapter 9

Good vs. bad models: Uncapacitated Facility Location

A set J of retailers has to be served by a set S of plants, yet to be built. We don't know where the plants will be, but there is a set I of potential sites, and there is

- a cost f_i for building plant $i \in I$
- a (transportation) cost c_{ij} from plant i to retailer j

Each retailer will be served by exactly one plant . Choose a subset S of I such that the total cost is minimized.

Variables:

- $x_i, i \in I$: 1 if plant i is built, 0 otherwise
- y_{ij} assigns retailer j to plant i: 1 if i serves retailer j, 0 otherwise

Good vs. bad models: Uncapacitated Facility Location

Objective function:

$$\sum_{i\in I} f_i x_i + \sum_{i\in I}\sum_{j\in J} c_{ij} y_{ij}$$

Constraints:

- for each customer, one facility: $\sum_{i\in I} y_{ij} = 1$
- customers go to mall i if it's there:

$$\sum_{j\in J} y_{ij} \le |J| x_i \quad \forall i \in I$$

or

$$y_{ij} \le x_i \quad \forall i \in I, j \in J$$

Good vs. bad models: Uncapacitated Facility Location

(A)

$$\begin{aligned} \min \quad & \sum_{i\in I} f_i x_i + \sum_{i\in I}\sum_{j\in J} c_{ij} y_{ij} \\ & \sum_{i\in I} y_{ij} = 1 \quad \forall i \in I \\ & \sum_{j\in J} y_{ij} \le |J| x_i \quad \forall i \in I \\ & x_i, y_{ij} \in \{0,1\} \end{aligned}$$

(B)

$$\begin{aligned} \min \quad & \sum_{i\in I} f_i x_i + \sum_{i\in I}\sum_{j\in J} c_{ij} y_{ij} \\ & \sum_{i\in I} y_{ij} = 1 \quad \forall i \in I \\ & y_{ij} \le x_i \quad \forall i \in I, j \in J \\ & x_i, y_{ij} \in \{0,1\} \end{aligned}$$

(A) and (B) are equivalent. However, for $|I| = |J| = 40$,

(A) takes 14 hours[1]

(B) takes 2 seconds

[1] On AMPL with an old version of CPLEX.

How do we solve an Integer Programming problem?

$$\begin{aligned} (P) \qquad \min \quad & cx \\ & Ax \ge b \\ & x_i \in \mathbb{Z}, i \in J \subseteq \{1, 2\dots, n\} \end{aligned}$$

- Relaxing integrality gives an LP problem (easy)
- Solving LP gives a lower bound
- But the solution x^* may have fractional component $x_i^* \notin \mathbb{Z}$, with $i \in J$ (for example, 3.31), infeasible for (P).

$\Rightarrow$ **Divide** the solution set, **partition** the problem into two new subproblems, P_1 and P_2, with

$$P_1 : x_i \le \lfloor x_i^* \rfloor \qquad P_2 : x_i \ge \lceil x_i^* \rceil$$

($P_1 : x_i \le 3$ and $P_2 : x_i \ge 4$) and recursively solve P_1 and P_2.

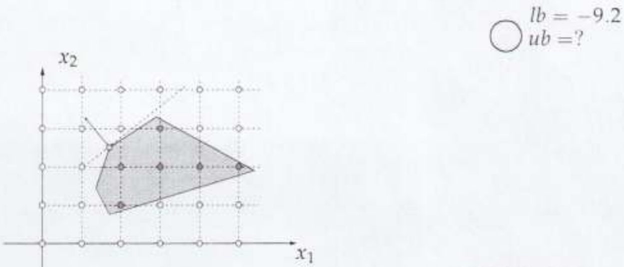
- Good: **no** feasible solution of P_1 or P_2 has $x_i = 3.31$

The Branch&Bound - Devide and concour

-

- If we solve the LP relaxation of P_1 and find a fractional point, we can recursively branch on P_1 and obtain two new subproblems P_3 and P_4.
- In principle, we have to branch on any node P_k unless its LP relaxation returns a feasible solution or it is infeasible.

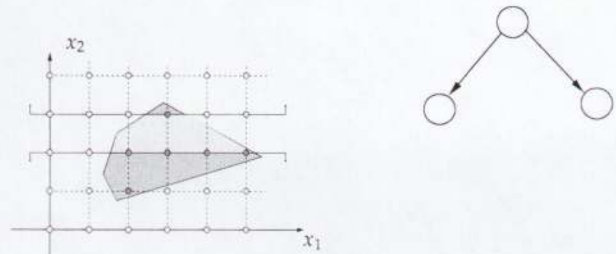
Example: minimize $11x_1 - 10x_2$



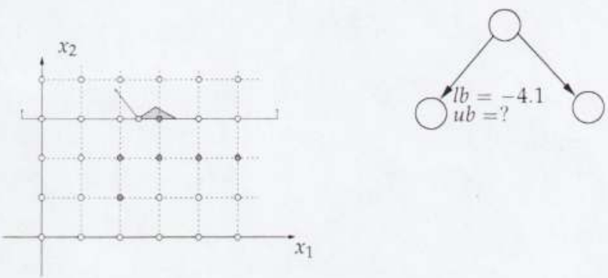
We can only cut out a branch when an upper bound of ~~a~~ another branch is smaller than its lower bound.

Thus the B&B used in 419 is not perfect.

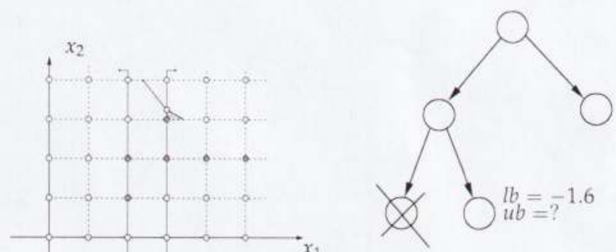
Example: minimize $11x_1 - 10x_2$



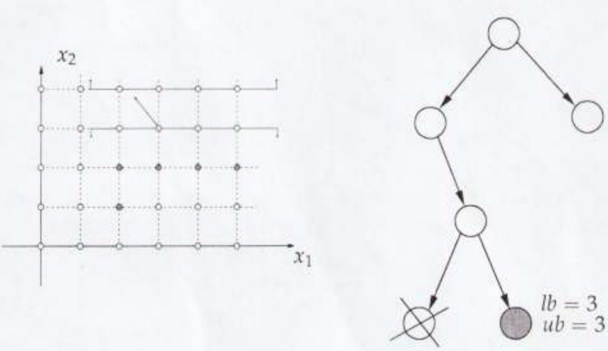
Example: minimize $11x_1 - 10x_2$



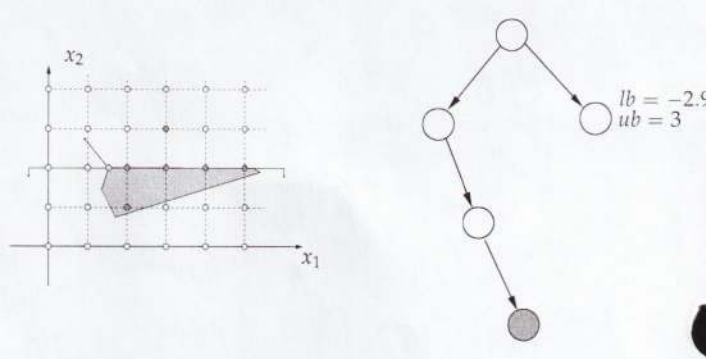
Example: minimize $11x_1 - 10x_2$



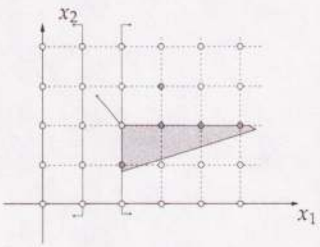
Example: minimize $11x_1 - 10x_2$

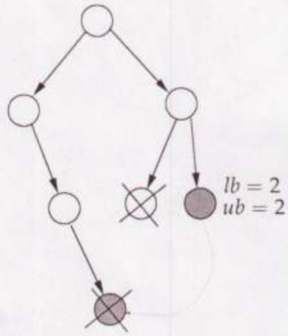


Example: minimize $11x_1 - 10x_2$



Example: minimize $11x_1 - 10x_2$





The **Bound** in Branch&Bound

In practice, the upper bound is very useful! Suppose you just found an **upper** bound of 194.

- P_3 has a **lower** bound of 146, P_4 of 203
- 203 is a lower bound for $P_4 \Rightarrow$ any feasible solution of P_4 has objective function value worse than 203 (i.e., ≥ 203)
- We already have something better (194) $\Rightarrow$ discard P_4

How to find upper bounds?

Example: Knapsack problem

$$\begin{array}{llllllll} & \min & 3x_1 & +5x_2 & +8x_3 & +6x_4 & +12x_5 & \\ (IP_0) & & 10x_1 & +7x_2 & +7x_3 & +5x_4 & +3x_5 & \geq 22 \\ & & x_1, & x_2, & x_3, & x_4, & x_5 & \in \{0,1\} \end{array}$$

The general Knapsack problem is:

$$\begin{array}{ll} \min & \sum_{i=1}^{n} w_i x_i \\ & \sum_{i=1}^{n} c_i x_i \geq C \\ & x_i \in \{0,1\} \qquad \forall i = 1, 2 \ldots, n \end{array}$$

Suppose that variables are sorted in non-decreasing order w.r.t. weight/value[2]. The LP relaxation is solved as follows:

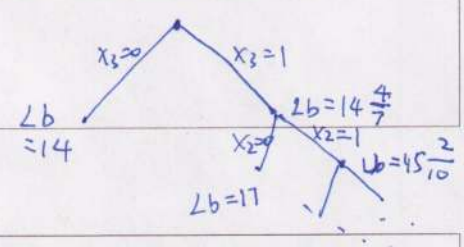
1. Set $R := C$
2. **for** $i : 1, 2 \ldots, n$
3. **if** $c_i \leq R$, **then** $x_i := 1$; $R := R - c_i$
4. **else** $x_i := R/c_i$; **stop**

[2]Those with a small weight/value ratio are more likely to be chosen.

Solve the LP relaxation

$$\begin{array}{llllllll} & \min & 3x_1 & +5x_2 & +8x_3 & +6x_4 & +12x_5 & \\ (LP_0) & & 10x_1 & +7x_2 & +7x_3 & +5x_4 & +3x_5 & \geq 22 \\ & & x_1, & x_2, & x_3, & x_4, & x_5 & \in [0,1] \end{array}$$

- Solve this by the greedy method.
- Solution: $x_0^* = (1, 1, \frac{5}{7}, 0, 0)$, $V_0^* = 13\frac{5}{7}$.
- V_0^* - the lower bound for the problem IP_0.
- Split the problem into two cases: $X_3 = 1$ (IP_1)and $X_3 = 0$ (IP_2) and consider the LP relaxations LP_1 and LP_2.



to make the tree asymmetric is better.

Consider subproblem

check like this

Solve LP_1 problem $X_3 = 1$ (we will consider LP_2 later)

$$\begin{array}{llllllll} & \min & 3x_1 & +5x_2 & +8 & +6x_4 & +12x_5 & \\ (LP_1) & & 10x_1 & +7x_2 & +7 & +5x_4 & +3x_5 & \geq 22 \\ & & x_1, & x_2, & & x_4, & x_5 & \in [0,1] \end{array}$$

- Solve this by the greedy method.
- Solution: $x_1^* = (1, \frac{5}{7}, 1, 0, 0)$, $V_1^* = 14\frac{4}{7}$.
- V_1^* - the lower bound for the problem IP_1.
- Split the problem into two cases: $X_2 = 1$ and $X_2 = 0$ and consider the LP relaxations LP_3 and LP_4.

Consider subproblem

Solve LP_3, $X_2 = 1$, $X_3 = 1$.

$$\begin{array}{llllllll} & \min & 3x_1 & +5 & +8 & +6x_4 & +12x_5 & \\ (LP_1) & & 10x_1 & +7 & +7 & +5x_4 & +3x_5 & \geq 22 \\ & & x_1, & & & x_4, & x_5 & \in [0,1] \end{array}$$

- Solve this by the greedy method.
- Solution: $x_3^* = (\frac{8}{10}, 1, 1, 0, 0)$, $V_3^* = 15\frac{2}{10}$.
- V_3^* - the lower bound for the problem IP_3.
- Split the problem into two cases: $X_2 = 1$ and $X_2 = 0$ and consider the LP relaxations L_3 and L_4.

Consider subproblem

Solve LP_3, $X_2 = 0$, $X_3 = 1$.

$$\begin{array}{rlllll} \min & 3x_1 & +8 & +6x_4 & +12x_5 & \\ (LP_1) & 10x_1 & +7 & +5x_4 & +3x_5 & \geq 22 \\ & x_1, & & x_4, & x_5 & \in [0,1] \end{array}$$

- Solve this by the greedy method.
- Solution: $x_4^* = (1, 0, 1, 1, 0)$, $V_4^* = 17$.
- V_4^* - the upper bound for the whole IP problem! Also is the optimal solution to IP_4.

Example: Knapsack problem

Solve LP_2 problem $X_3 = 0$

$$\begin{array}{rllllll} \min & 3x_1 & +5x_2 & +6x_4 & +12x_5 & \\ (LP_1) & 10x_1 & +7x_2 & +5x_4 & +3x_5 & \geq 22 \\ & x_1, & x_2, & x_4, & x_5 & \in [0,1] \end{array}$$

- Solve this by the greedy method.
- Solution: $x_2^* = (1, 1, 0, 1, 0)$, $V_2^* = 14$.
- V_2^* is the new upper bound for the entire problem!! Also optimal solution to IP_2.
- Key observation: lower bound for IP_3 is bigger than upper bound for the entre problem. DONE!!!!!

Branch&Bound

- $z^{\text{ub}} = +\infty$
- $\mathcal{L} \leftarrow \{P\}$
- **while** $\mathcal{L} \neq \emptyset$

  Choose P' from $\mathcal{L}$ and set $\mathcal{L} = \mathcal{L} \setminus \{P'\}$

  Relax $P' \rightarrow$ obtain LP'

  solve LP', obtain solution $x^{LP'}$ and lower bound $z^{LP'}$

  look for solution feasible for P', obj. $z^{P'}$

  if $z^{P'} < z^{\text{ub}}$, set $z^{\text{ub}} \leftarrow z^{P'}$

  if $z^{LP'} < z^{\text{ub}}$ and $x^{LP'}$ infeasible for P

    choose $x_i : x_i^{LP'} \notin \mathbb{Z}$

    create $P'' : x_i :\leq \lfloor x_i^{LP'} \rfloor$

    create $P''' : x_i :\geq \lceil x_i^{LP'} \rceil$

    $\mathcal{L} \leftarrow \mathcal{L} \cup \{P'', P'''\}$

Formulation

$y=1 \iff (x_{ki}=1) \wedge (x_{lj}=1)$.

- The objective function is nonlinear, but we know what to do...
- Introduce a new variable y_{kilj} defined as $x_{ki}x_{lj}$ binary.

$\Rightarrow$ y_{kilj} are binary too, and are subject to the constraints:

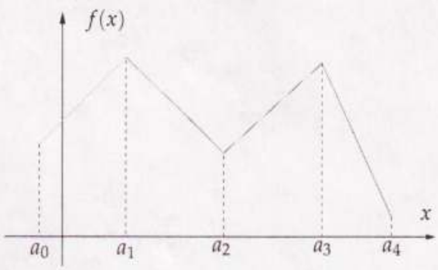
$$\begin{cases} y_{kilj} & \le x_{lj} \\ y_{kilj} & \le x_{ki} \\ y_{kilj} & \ge x_{ki} + x_{lj} - 1 \end{cases}$$

Formulation

$$\begin{array}{lll} \min & \sum_{k\in N}\sum_{l\in N}\sum_{i\in N}\sum_{j\in N} & f_{kl}d_{ij}y_{kilj} \\ & \sum_{j\in N} x_{ij} = 1 & \forall i \in N \\ & \sum_{i\in N} x_{ij} = 1 & \forall j \in N \\ & y_{kilj} \le x_{ki} & \forall (k,i,l,j) \in N^4 \\ & y_{kilj} \le x_{lj} & \forall (k,i,l,j) \in N^4 \\ & y_{kilj} \ge x_{ki} + x_{lj} - 1 & \forall (k,i,l,j) \in N^4 \\ & y_{kilj} \in \{0,1\} & \forall (k,i,l,j) \in N^4 \\ & x_{ij} \in \{0,1\} & \forall (i,j) \in N^2 \end{array}$$

Piecewise Linear functions

Consider a univariate, piecewise linear function $f(x)$ made of n linear pieces.



- it can be modeled with linear constraints
- but the function is not convex, hence we need a MILP model this time

A model for piecewise linear functions

Variable x needs to be modeled depending on the a_i's.

- If $a_2 \le x \le a_3$, we want $f(x)$ to be between $f(a_2)$ and $f(a_3)$
- If $x = \lambda a_2 + (1-\lambda)a_3$, with $0 \le \lambda \le 1$, then $f(x)$ must be $\lambda f(a_2) + (1-\lambda)f(a_3)$
- In general, use variables λ_i: $x = \sum_{i=0}^{n} \lambda_i a_i$, where
  - only two λ_i's are non zero, **and**
  - they sum up to one, **and**
  - they are consecutive

e.g. to model x exactly at the midpoint between a_2 and a_3, we need $\lambda_2 = \lambda_3 = \frac{1}{2}$ and $\lambda_0 = \lambda_1 = \lambda_4 = 0$

A model for piecewise linear functions

OK, but how do we ensure the "only two" and the "consecutive" things?

$\Rightarrow$ with binary variables!

- Define one binary variable y_i for each linear piece:
- There is only one nonzero y_i
- y_i is 1 if x is between a_{i-1} and a_i 边界情况
- That is, we want
  - if $\lambda_0 > 0$, then $y_1 = 1$
  - if $\lambda_i > 0$ with $i = 1, 2 \ldots, n-1$, then $y_i = 1$ or $y_{i+1} = 1$
  - if $\lambda_n > 0$, then $y_n = 1$

each a_i has a λ_i.

$\lambda_i f(a_i) + \lambda_{i+1} f(a_{i+1})$

$\lambda_{i+1} f(a_{i+1}) + \lambda_i f(a_i)$

A model for piecewise linear functions

Introduce a new variable φ for $f(x)$. We have:

the function.

$$\begin{array}{rll} \varphi = & \sum_{i=0}^{n} \lambda_i f(a_i) & \\ x = & \sum_{i=0}^{n} \lambda_i a_i & \\ \sum_{i=0}^{n} \lambda_i = & 1 & \\ \sum_{i=1}^{n} y_i = & 1 & \\ \lambda_0 \le & y_1 & \\ \lambda_n \le & y_n & \\ \lambda_i \le & y_i + y_{i+1} & \forall i = 1, 2 \ldots, n-1 \\ \lambda_i \in & [0,1] & \forall i = 0, 2 \ldots, n \\ y_i \in & \{0,1\} & \forall i = 1, 2 \ldots, n \end{array}$$

约束 $\lambda_i \leftrightarrow \lambda_{i+1}$ 的连续

$a_i \to a_{i+1}$

约束

ISE 426
Optimization models and applications

Lecture 20 — November 10, 2015

- Bin Packing Problem
- Cutting Stock Problem
- Column generation

The bin packing problem

- Given a set of N bins of volume V and n_i objects of volumes v_i, $i = 1, \dots, n$.
- We want to pack the objects into bins using as few bins as possible.
- Consider $V = 11$, $n = 3$ from which $n_1 = 20$ objects have $v_1 = 5$, $n_2 = 10$ objects have $v_2 = 4$ and $n_3 = 9$ objects have $v_3 = 2$;
- Let us try a gready heristic, we get: 10 bins with two $(5,5)$ objects, 5 bins with $(4,4,2)$ objects and 1 bin with $(2,2,2,2)$ objects.
- Clearly 9 bins with $(5,4,2)$, 5 bins with $(5,5)$ and one bin with $(5,4)$ is better.

How do we model this as an Optimization model?

- y_i is a binary variable that indicates if a bin i is being used.
- x_{ij} is an integer variable that indicates how many objects of size j has been assigned to bin i.
- Clearly $x_{ij} \le My_i$.

$$
\begin{aligned}
\min \quad & \textstyle\sum_{i=1}^{N} y_i \\
& \textstyle\sum_{j=1}^{n} v_j x_{ij} \le V y_i \\
& \textstyle\sum_{i=1}^{N} x_{ij} = n_j \quad \forall j = 1 \dots, n \\
& x_{ij} \in \mathbf{Z} \quad \forall i = 1, \dots, N,\ j = 1 \dots, n \\
& y_i \in \{0,1\} \quad \forall i = 1, \dots, N
\end{aligned}
$$

Weak formulation, too much symmetry, each bin is the same.
May help to add constraints

$$y_1 \ge y_2 \ge y_3 \ge \dots \ge y_N$$

(Use the first bin first)

Formulation #2: extended formulation

find all good possible patterns

- Consider all sets s_k of objects (patterns) that can fit into one bin.
- That is, $(5,5)$, $(5,4,2)$ $(4,4,2)$, $(4,2,2,2)$, $(2,2,2,2,2)$. (We do not need to consider $(5,4)$ because it is dominated by $(5,4,2)$)
- Let S be the set of all feasible patterns s_k, $|S| = K$.

→ if the # is huge, it's not good formulation.

- x_k is an integer variable indicating how many bins are filled with pattern s_k, $s_k \in S$.
- Let a_{jk} be the number of times object of size v_j appears in s_k, for instance for $s_k = (4,4,2)$, for $v_j = 4$, $a_{jk} = 2$ and for $v_j = 2$, $a_{jk} = 1$.

$$
\begin{aligned}
\min \quad & \textstyle\sum_{k=1}^{K} x_k \\
& \textstyle\sum_{k=1}^{K} a_{jk} x_k \ge n_j \ \forall j = 1, \dots, n \\
& x_k \in \mathbf{Z} \quad \forall k = 1, \dots, K
\end{aligned}
$$

This is a "strong" formulation. The LP relaxation gives very good lower bounds.

b_{in}

Relaxation and another interpretation - cutting stock problem

choose some patterns → calculate dual → choose

$$
\begin{aligned}
\min \quad & \textstyle\sum_{k=1}^{K} x_k \\
& \textstyle\sum_{k=1}^{K} a_{jk} x_k \ge n_j \quad \forall j = 1, \dots, n \\
& x_k \ge 0 \quad \forall k = 1, \dots, K
\end{aligned}
$$

- Given very long (infinite) roll of paper (or steel) of width V we need to cut paper into pieces of length n_j and width v_j, $j = 1, \dots, n$.
- We are allowed to cut correct width, but smaller length and "glue" different lengths to obtain the right one.
- We want to use as little paper as possible.

# of patterns K huge

$K \times n$

$A = ($

Dual formulation

A^T k

$$
\begin{aligned}
\max \quad & \textstyle\sum_{j=1}^{n} n_j y_j \\
& \textstyle\sum_{j=1}^{n} a_{jk} y_j \le 1 \ \forall k = 1, \dots, K \\
& y_j \ge 0 \ \forall j = 1, \dots, n
\end{aligned}
$$

- Each pattern corresponds to variable x_k, which in turn, corresponds to a constraint

$$\sum_{j=1}^{n} a_{jk} y_j \le 1.$$

- Remember that given a basic feasible solutions we have a lot of $x_k = 0$.
- If the dual constraint for a given k is not feasible, then the corresponding x_k should be, possibly, nonzero.
- Column generation technique - generate k's for which

$$\sum_{j=1}^{n} a_{jk} y_j > 1$$

$\}^{n}$

Column generation

$$
\begin{aligned}
\max\quad & \textstyle\sum_{j=1}^{n} n_j y_i \\
& \textstyle\sum_{j=1}^{n} a_{jk} y_j \le 1 \ \forall k = 1, \ldots, K \\
& y_j \ge 0 \ \forall j = 1, \ldots, n
\end{aligned}
$$

- Start with a few patterns and variables x_k (the rest of $x_k = 0$).
- Solve the primal problem with only those patterns.
- Compute the corresponding dual solution
- Column generation technique - generate k's for which

> 0 $\rightarrow x \geq 0$

slackness $\sum_{j=1}^{n} a_{jk} y_j > 1$ to find

and include them into the primal problem.

column the reduced cost

to find $x_k > 0$ is good.

Column generation

$$\sum_{j=1}^{n} a_{jk} y_j > 1$$

How do we find new k? Find k for which $\sum_{j=1}^{n} a_{jk} y_j$ is the largest.

$$
\begin{aligned}
\max\quad & \textstyle\sum_{j=1}^{n} a_{jk} y_j \\
& \textstyle\sum_{j=1}^{n} v_j a_{jk} \le V \ \forall k = 1, \ldots, K \\
& a_{jk} \in \mathbf{Z} \ \forall j = 1, \ldots, n
\end{aligned}
$$

The knapsack problems!!

Practice problems for the quiz

Reformulate using constraints of a Linear Programming problem or, if necessary, Mixed Integer Linear Programming Problem.

1. $\max\{|x|, |y|\} \le 1$.(6pts)
2. $\max\{|x|, |y|\} \ge 1$.(6pts)

Formulation, Mixed Integer/Goal Programming

Kyra is organizing a large dinner party. There are k tables, each sitting n people. There are m men attending and w women. She needs to assign seats at the tables in such a way that the number of men and women at each table does not differ by more than two. Formulate this as a feasible set of an integer linear programming problem.
Is the above problem *always* feasible? Explain.
Write an integer linear optimization problem that minimizes the number of tables that violate the condition on the maximum difference between the number of men and women.

Reformulation using binary variables

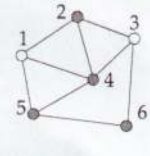
Consider a set of vectors $x \in R^n$ described by the following conditions.

$$\max_i \{x_1, x_2, x_3, \ldots, x_n\} \ge 1$$

Describe this set by using a set of linear constraints and binary variables, as we did in homework and class. In other words, you only should use linear constraints that can involve continuous and/or binary variables, and all feasible solutions for this set of constraints should give x that is feasible for the the above set and vice versa.

Integer Programming

Consider a graph $G = (V, E)$, and a cost C_{ij} for each edge $\{i, j\} \in E$. Suppose you want to find the subset S of V with at least k nodes, such that the cost of all edges, that link a node in S with a node outside of S is minimized. For example, if S is the set of four dark nodes is the graph below, then the total cost is $C_{12} + C_{14} + C_{15} + C_{23} + C_{34} + C_{36}$



1. Consider binary variables that indicate if a node is in S or not. Consider also binary variables that indicate if an edge is connecting a node in S with a node outside of S. Now write down conditions between these types of variables, which ensure logical implications: for all $\{i, j\} \in E$, if $i \in S$ and $j \in V/S$ then edge $\{i, j\}$ connects node in S with a node outside S.
2. Write the full formulation of the problem of selecting at least k nodes so that the edge cost is minimized.

ISE 426
Optimization models and applications

Lecture 21 — November 17, 2015

- Intro to Stochastic Programming (SP)

Reading:

- Book by Kall & Wallace (pdf – Chapter 1 up to 1.6)
- J.R. Birge, F. Louveaux, Stochastic Programming

Decision problems under uncertainty

In all problems we've seen so far, we assumed 100% knowledge of the parameters (production capacity, customer demand, etc.)

In this context, a global solution to an Optimization problem is known to be the best possible thing to do

However, perfect knowledge of the parameters is unrealistic:

- We don't know what our competitor, customer, even co-worker or boss, will decide
- Nature doesn't usually tell us in advance what it will do: e.g., weather is never certain
- Parameters are often estimated, i.e., given with a level of accuracy < 100%

Decision problems under uncertainty

Important: both the parameters and the variables are unknown in advance. However,

- the model's variables are something **we** decide

i.e. **we** find the right ones if we have the right tools

- the model's parameters are **not** under our control: if we treated them as variables, we'd find the ideal (and unrealistic) situation

e.g. the competitor goes bankrupt AND the temperature stays good all winter AND our employees decide to cut their salary AND we win the lottery AND…

Example: the uncertain knapsack problem

At a flea market in Rome, you spot n objects (old pictures, a vessel, rusty medals…) that you could re-sell in your antique shop for about double the price.

- You want these objects to pay for your flight ticket to Rome, which cost C.
- Also, your backpack can carry all of them, but you don't want it heavy, so you want to buy the objects that will load your backpack as little as possible.

How do you solve this problem?

Example: the uncertain knapsack problem

Each object $i = 1, 2, \ldots, n$ has a price p_i and a weight w_i.

- Variables: one variable x_i for each $i = 1, 2 \ldots, n$. This is a "yes/no" variable, i.e., either you take the i-th object or not.
- Constraint: total revenue must be at least C

  The revenue for the i-th item can be between $0.8p_i$ and $1.1p_i$, or $\alpha_i p_i$, with $0.8 \leq \alpha_i \leq 1.1$
- Objective function: the total weight

Example: the uncertain knapsack problem

$$
\begin{array}{ll}
\min & \sum_{i=1}^n w_i x_i \\
 & \sum_{i=1}^n \alpha_i p_i x_i \geq C \\
 & x_i \in \{0, 1\} \quad \forall i = 1, 2, \ldots, n
\end{array}
$$

- If α_i were variables, our solver would find an optimal solution where $\alpha_i = 1.1$ for all i.
- α_i is **not** a variable – it's an unknown parameter!
- Robust optimization - worst case scenario: simply set all α_is to 0.8 and solve the problem.
- We may look at each object and see the usual value of α_i for that object.

How to handle uncertainty?

Handling all sources of uncertainty is almost impossible, and we better model and simplify them.

- scenarios: enumerate a few "situations" that may happen and solve a problem for each of them

⇒ each scenario gives us some insight on what goes wrong and how to tackle it

- stages: some decisions have to be made now, some others may be made later
- however, the later decisions are influenced by what we decide now (and the events between "now" and "later")

⇒ We have to model that influence, too – it's as if we are anticipating the later decisions by taking into account what happens in the meantime

Farmer excercise

SP solves optimization problems with stochastic info.

- You grow **wheat**, **corn**, and **sugar beet** on 500 acres of land
- You should decide how many acres to use for each crop
- Planting an acre costs \$150, \$230, and \$260, respectively
- Need at least 200t of wheat and 240t of corn for cattle feed
- Eccess production is sold at \$170/t and \$150/t, resp.
- If less is produced, it is bought at \$238/t and \$210/t, resp.
- Sugar beet sells at \$36/t up to 6000t, and at \$10/t above that quota

The **average** yields of crop are:

- 2.5t/acre for wheat
- 3t/acre for corn
- 20t/acre for beet

Depending on how good the weather is, these yields may decrease or increase by 20%. How to solve this problem?

Model

Variables:

x_1: acres for growing wheat
x_2: acres for growing corn
x_3: acres for growing (sugar) beet
w_1: tons of wheat exceeding 200t (to be sold)
w_2: tons of corn exceeding 240t (to be sold)
w_3: tons of beet below 6,000t (to be sold at 36\$/t)
w_4: tons of beet above 6,000t (to be sold at 10\$/t)
y_1: tons of wheat to be bought (when $x_1 < 200$t)
y_2: tons of corn to be bought (when $x_2 < 240$t)

Objective function (to be maximized):

$$\begin{array}{lr} 170w_1 + 150w_2 + 36w_3 + 10w_4 & \text{crop sale} \\ -(150x_1 + 230x_2 + 260x_3) & \text{planting costs} \\ -(238y_1 + 210y_2) & \text{purchased wheat/corn} \end{array}$$

Model

Constraints:

$$\begin{array}{lr} x_1 + x_2 + x_3 \leq 500 & \text{total area} \\ w_1 \leq 2.5x_1 - 200 + y_1 & \text{excess wheat} \\ w_2 \leq 3x_2 - 240 + y_2 & \text{excess corn} \\ 20x_3 = w_3 + w_4 & \text{total beet} \\ y_1 \geq 200 - 2.5x_1 & \text{purchased wheat} \\ y_2 \geq 240 - 3x_2 & \text{purchased corn} \\ w_3 \leq 6,000 & \text{bound on the quota} \\ x_1, x_2, x_3, w_1, w_2, w_3, w_4, y_1, y_2 \geq 0 & \end{array}$$

Optimal solution: $(x_1, x_2, x_3) = (120, 80, 300)$, with $w_1 = 100$, $w_3 = 6,000$, and a total profit of 118,600\$.

Model under uncertainty

Scenario #1: yields below average (−20%).
Constraints (only those that change):

$$\begin{array}{lr} w_1 \leq 2x_1 - 200 + y_1 & \text{excess wheat} \\ w_2 \leq 2.4x_2 - 240 + y_2 & \text{excess corn} \\ 16x_3 = w_3 + w_4 & \text{total beet} \\ y_1 \geq 200 - 2x_1 & \text{purchased wheat} \\ y_2 \geq 240 - 2.4x_2 & \text{purchased corn} \end{array}$$

Optimal solution: $(x_1, x_2, x_3) = (100, 25, 375)$, with $w_3 = 6,000$, $y_2 = 180$, and a total profit of 59,950\$.

Scenario #2: average yields. Same solution as before (118,600\$).

Model under uncertainty

Scenario #3: yields above average (+20%).
Constraints (only those that change):

$$\begin{array}{lr} w_1 \leq 3x_1 - 200 + y_1 & \text{excess wheat} \\ w_2 \leq 3.6x_2 - 240 + y_2 & \text{excess corn} \\ 24x_3 = w_3 + w_4 & \text{total beet} \\ y_1 \geq 200 - 3x_1 & \text{purchased wheat} \\ y_2 \geq 240 - 3.6x_2 & \text{purchased corn} \end{array}$$

Optimal solution: $(x_1, x_2, x_3) = (183.3, 66.7, 250)$, with $w_1 = 350$, $w_3 = 6,000$, and a total profit of 167,666.67\$.

Which one is the right solution?

The pessimistic would choose the solution of scenario #1.

- At least we're sure we'll make at least 59,950$.
- If the weather turns out to be OK, the "pessimistic" choice gives us a return of 86,600$. → use $\vec{x}$ in scenario #1 in ave model get!
- If the weather is great, we'd make 113,250$.

The neither lucky nor unlucky would go for scenario #2.

- If the weather is bad, we make 55,120$.
- If the weather is OK, we make 118,600$.
- In the best case, we make 148,000$.

Those feeling lucky would choose the solution of scenario #3.

- If the weather is bad, we make 47,700$.
- If the weather is OK, we make 107,683.33$.
- In the best case, we make 167,666.67$.

Stochastic approach

We have nine values for the profit, and don't really know which one makes more sense.

Suppose that each scenario has the same probability: $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$.
How can we get a solution that considers both three scenarios?

- The decision we want to make **now**, i.e., the area for wheat, corn, and beet (x_1, x_2, and x_3) are still variables
- For those quantities that depend on Nature, and that we have no control on, there is one variable for each scenario:

e.g. for w_1 there are three new variables w_{11}, w_{12}, and w_{13}

- each w_{1j} is the exceeding wheat if scenario j is realized
- these are second stage variables, in that their value is a consequence of the decisions of Nature, but they still influence our decision today.

How does the model change?

It clearly gets more complicated…
Variables (j is a scenario, $j \in \{1, 2, 3\}$):

- x_1: acres for growing wheat
- x_2: acres for growing corn
- x_3: acres for growing (sugar) beet
- w_{1j}: tons of wheat exceeding 200t (to be sold)
- w_{2j}: tons of corn exceeding 240t (to be sold)
- w_{3j}: tons of beet below 6,000t (to be sold at 36$/t)
- w_{4j}: tons of beet above 6,000t (to be sold at 10$/t)
- y_{1j}: tons of wheat to be bought (when $x_1 < 200$t)
- y_{2j}: tons of corn to be bought (when $x_2 < 240$t)

Model

Objective function (to be maximized):

$$\begin{array}{ll} & -(150x_1 + 230x_2 + 260x_3) \\ +\frac{1}{3} & [(170w_{11} + 150w_{21} + 36w_{31} + 10w_{41}) - (238y_{11} + 210y_{21})] \\ +\frac{1}{3} & [(170w_{12} + 150w_{22} + 36w_{32} + 10w_{42}) - (238y_{12} + 210y_{22})] \\ +\frac{1}{3} & [(170w_{13} + 150w_{23} + 36w_{33} + 10w_{43}) - (238y_{13} + 210y_{23})] \end{array}$$

Constraints:

$$\begin{array}{ll} x_1 + x_2 + x_3 \leq 500 & \text{total area} \\ w_{11} \leq 2x_1 - 200 + y_{11} & \text{excess wheat} \\ w_{12} \leq 2.5x_1 - 200 + y_{12} & \\ w_{13} \leq 3x_1 - 200 + y_{13} & \\ w_{21} \leq 2.4x_2 - 240 + y_{21} & \text{excess corn} \\ w_{22} \leq 3x_2 - 240 + y_{22} & \\ w_{23} \leq 3.6x_2 - 240 + y_{23} & \end{array}$$

Stochastic Programming model (cont'd)

Constraints:

$$\begin{array}{ll} 16x_3 = w_{31} + w_{41} & \text{total beet} \\ 20x_3 = w_{32} + w_{42} & \\ 24x_3 = w_{33} + w_{43} & \\ y_{11} \geq 200 - 2x_1 & \text{purchased wheat} \\ y_{12} \geq 200 - 2.5x_1 & \\ y_{13} \geq 200 - 3x_1 & \\ y_{21} \geq 240 - 2.4x_2 & \text{purchased corn} \\ y_{22} \geq 240 - 3x_2 & \\ y_{23} \geq 240 - 3.6x_2 & \\ w_{3j} \leq 6,000 & \forall j \in \{1, 2, 3\} \\ x_1, x_2, x_3 \geq 0 & \\ w_{1j}, w_{2j}, w_{3j}, w_{4j}, y_{1j}, y_{2j} \geq 0 & \forall j \in \{1, 2, 3\} \end{array}$$

Solution

$(x_1, x_2, x_3) = (170, 80, 250)$. Expected profit: 108,390$.

scenario	w_1	w_2	w_3	w_4	y_1	y_2	profit
1	140	0	4,000	0	0	48	48,820$
2	225	0	5,000	0	0	0	109,350$
3	310	48	6,000	0	0	0	167,000$

What do 108,390$ mean? If the farmer repeated this choice for the next n years, under the same conditions he would make, on average, 108,390$ a year.

Why is it not equal to 118,600$, the profit in scenario 2? Because we considered uncertainty.

Averages and forecasts

What if we averaged the (deterministic) profits from the three scenarios?

$$\frac{59,950\$ + 118,600\$ + 167,666.67\$}{3} \approx 115,406\$,$$

yet another number. That would be the average profit over n years if we knew in advance, each year, how the weather would be and choose x_1, x_2, x_3 consequently.

The difference $115,406\$ - 108,390\$ = 7,016\$$ is called *Expected Value of Perfect Information* (EVPI).

- It's what we "lose" for being realistic...
- It's what we'd gain from **knowing** the weather before planting.

Averages and forecasts, 2

If we were lazy and considered **only** scenario #2, we could just sit and watch the average over n years. We'd have a profit of

- $55,120\$$ in bad years
- $118,600\$$ in OK years
- $148,000\$$ in very good years

Given that they occur with **known** probability $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$, we'd observe an average profit of

$$\frac{55120\$ + 118,600\$ + 148,000\$}{3} \approx 107,240\$,$$

which is clearly below $115,406\$$ because we bet on OK weather every year instead of using perfect knowledge of the weather.

The difference $108,390\$ - 107,240\$ = 1,150\$$ is the *value of the stochastic solution* (VSS). It tells you how much you gain if you use Stochastic Programming.

To recap

- Case #1: We know all parameters (the weather every year). We have perfect information and make the decision with a single, deterministic optimization model. We make a lot of money (in our dreams). [115,406$]
- Case #2: We don't know the parameters, but we pick the average value and solve a deterministic model. The solution is optimal only assuming those values of the parameters, which won't occur all the time. We won't make a lot of money. [107,240$]
- Case #3: We don't know the parameters, but we formulate a model that considers all possible events and their impact on our solution. [108,390$]

Note: we did make an assumption though. We assumed to know the **probabilities** of the three scenarios.

A more prudent farmer - robust optimization

...maximizes the minimum return instead of the expected value. Objective function (to be maximized):

$$\begin{array}{ll} & -(150x_1 + 230x_2 + 260x_3) \\ +\min & [(170w_{11} + 150w_{21} + 36w_{31} + 10w_{41}) - (238y_{11} + 210y_{21}), \\ & (170w_{12} + 150w_{22} + 36w_{32} + 10w_{42}) - (238y_{12} + 210y_{22}), \\ & (170w_{13} + 150w_{23} + 36w_{33} + 10w_{43}) - (238y_{13} + 210y_{23})] \end{array}$$

Optimal solution: $(x_1, x_2, x_3) = (100, 25, 375)$, with $w_3 = 6,000$, $y_2 = 180$, and a total profit of 59,950$.

- Incidentally, the same solution as assuming scenario #1.
- Not always the case: here we are looking at the worst-case scenario as it would arise from the second stage variables,
- and it turns out that the bad weather scenario is the worst-case scenario (obvious)

A quick comparison of RO and SP

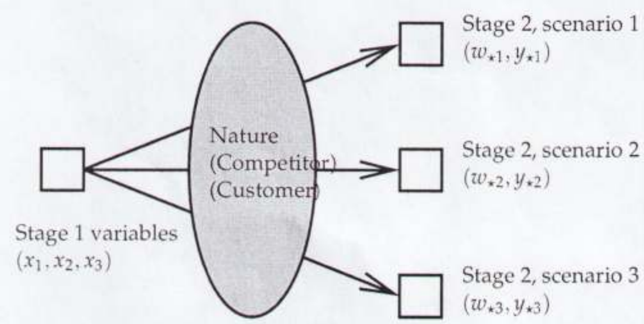
Robust Optimization:

- Robust optimization considers a set of possible parameter values and optimizes the **worst-case scenario**
- Gives a guarantee that the outcome will not be worse then estimted.
- No need for probability (distribution) estimates.
- Can be too pessimistic if all possible values are considered.
- Models may get more complex, but not larger.

Stochastic Porgramming:

- Optimizes average case.
- Needs probability estimtes.
- Not as pessimistic.
- Usually does not complicate type of models, but makes them much larger.

Scenarios and stages



Review: EVPI

The *Expected Value of Perfect Information* is the difference between what you'd make **with no uncertainty** and what you expect to make – the solution of an SP.

- Bad weather? Plant $(100, 25, 375)$ and make a 59,950$ profit
- OK weather? Plant $(120, 80, 300)$, make a 118,600$ profit
- Good weather? Plant $(183.3, 66.7, 250)$, make 167,666.67$
- The three scenarios have probability of $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$

⇒ On average, we make $\frac{1}{3}59,950\$ + \frac{1}{3}118,600\$ + \frac{1}{3}167,666.67\$ = 115,406\$$

- SP tells us we make 108,390$

⇒ $\text{EVPI} = 115,406\$ - 108,390\$ = 7,016\$$

Review: VSS

Value of Stochastic Solution: the difference between the solution of an SP program and the expected value of the objective function when we fix parameters to **average** values and use the corresponding optimal solution.

- Assume weather will be OK all the time

⇒ Plant $(120, 80, 300)$, no matter what

- Bad weather? Make a 55,120$ profit
- OK weather? Make a 118,600$ profit
- Good weather? Make 148,000$
- The three scenarios have probability of $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$

⇒ On average, we make $\frac{1}{3}55,120 + \frac{1}{3}118,600 + \frac{1}{3}148,000\$ = 107,240\$$

- SP tells us we make 108,390$

⇒ $\text{VSS} = 108,390\$ - 107,240 = 1,150\$$

From deterministic...

Consider a set S of scenarios, a set of n first stage variables x and a set of p second stage (or recourse) variables, y.
We have to turn the *deterministic* model,

$$\begin{array}{lllll} \min & c_1x_1 & \ldots + c_nx_n & +d_1y_1 & \ldots + d_py_p \\ & a_{11}x_1 & \ldots + a_{1n}x_n & +b_{11}y_1 & \ldots + b_{1p}y_p \leq f_1 \\ & a_{21}x_1 & \ldots + a_{2n}x_n & +b_{21}y_1 & \ldots + b_{2p}y_p \leq f_2 \\ & \vdots & & & \\ & a_{m1}x_1 & \ldots + a_{mn}x_n & +b_{m1}y_1 & \ldots + b_{mp}y_p \leq f_m, \end{array}$$

into a **stochastic** due to uncertain parameters (c, d, a, b, f).

- First stage variables are decisions to be made now, regardless of the scenario that will actually be realized.
- Recourse variables represent decisions to be made after part of the uncertainty is revealed.

...to Stochastic models

Introduce a set of recourse variables $y_1^s \ldots y_p^s$ for each $s \in S$.
Consider an uncertain value of the parameter: instead of the (c, d, a, b, f) we have one $(c^s, d^s, a^s, b^s, f^s)$ for each $s \in S$.
If every $s \in S$ has probability p_s, the **expected value** of the objective function is

$$\sum_{s \in S} p_s(c_1^s x_1 \ldots + c_n^s x_n + d_1^s y_1^s \ldots + d_p^s y_p^s)$$

And we rewrite all constraints as

$$\begin{array}{llllll} a_{11}^s x_1 & \ldots + a_{1n}^s x_n & +b_{11}^s y_1^s & \ldots + b_{1p}^s y_p^s \leq f_1^s & \forall s \in S \\ a_{21}^s x_1 & \ldots + a_{2n}^s x_n & +b_{21}^s y_1^s & \ldots + b_{2p}^s y_p^s \leq f_2^s & \forall s \in S \\ \vdots & & & & \\ a_{m1}^s x_1 & \ldots + a_{mn}^s x_n & +b_{m1}^s y_1^s & \ldots + b_{mp}^s y_p^s \leq f_m^s & \forall s \in S \end{array}$$

Example: A facility location problem

A company wants to open a few malls, choosing from a set I of potential locations, to serve a set J of customers (towns).

- Each mall i, if open, has a capacity of p_i
- There is a transportation cost d_{ij} between $i \in I$ and $j \in J$
- Building a mall at i costs c_i
- Each town $j \in J$ has a demand f_j to be satisfied by **one** of the (open!) facilities
- A demand not served costs the company g per unit

Deterministic model

Variables:

- x_i: open a mall at i
- y_{ij}: mall i serves town j
- z_i unsatisfied demand for mall i

Constraints:

- Town j is served by mall i if mall i is open: $y_{ij} \leq x_i$, for all malls $i \in I$ and towns $j \in J$
- Town $j \in J$ is served by one mall: $\sum_{i \in I} y_{ij} = 1$ for all $j \in J$
- Definition of variable z_i, i.e., demand that mall i does not satisfy: $z_i \geq \sum_{j \in J} f_j y_{ij} - p_i$
- $z_i \geq 0 \forall i \in I$; $\quad x_i \in \{0, 1\} \forall i \in I$; $\quad y_{ij} \in \{0, 1\} \forall i \in I, j \in J$

Objective function: $\sum_{i \in I} c_i x_i + g \sum_{i \in I} z_i + \sum_{i \in I} \sum_{j \in J} d_{ij} y_{ij}$

Deterministic model

$$
\begin{array}{lll}
\min & \sum_{i\in I} c_i x_i + g\sum_{i\in I} z_i + \sum_{i\in I}\sum_{j\in J} d_{ij} y_{ij} & \\
& y_{ij} \le x_i & \forall i \in I, \forall j \in J \\
& \sum_{i\in I} y_{ij} = 1 & \forall j \in J \\
& z_i \ge \sum_{j\in J} f_j y_{ij} - p_i & \forall i \in I \\
& z_i \ge 0,\ \ x_i \in \{0,1\} & \forall i \in I \\
& y_{ij} \in \{0,1\} & \forall i \in I, j \in J
\end{array}
$$

Suppose the demand f_i is not known, and is assumed of three types (*scenarios*): f_i^A, f_i^B, and f_i^C, with probabilities p^A, p^B, and p^C.

- x_i are first-stage variables: a decision to be made now
- z_i are recourse variables: they depend on actual demand
- y_{ij} could be both! Let's assume they are chosen by the population according to demand $\Rightarrow$ recourse variables

Stochastic model I

$$
\begin{array}{lll}
\min & \sum_{i\in I} c_i x_i + & \\
& +p^A(g\sum_{i\in I} z_i^A + \sum_{i\in I}\sum_{j\in J} d_{ij} y_{ij}^A) & \\
& +p^B(g\sum_{i\in I} z_i^B + \sum_{i\in I}\sum_{j\in J} d_{ij} y_{ij}^B) & \\
& +p^C(g\sum_{i\in I} z_i^C + \sum_{i\in I}\sum_{j\in J} d_{ij} y_{ij}^C) & \\
& y_{ij}^A \le x_i & \forall i \in I, \forall j \in J \\
& y_{ij}^B \le x_i & \forall i \in I, \forall j \in J \\
& y_{ij}^C \le x_i & \forall i \in I, \forall j \in J \\
& \sum_{i\in I} y_{ij}^A = 1 & \forall j \in J \\
& \sum_{i\in I} y_{ij}^B = 1 & \forall j \in J \\
& \sum_{i\in I} y_{ij}^C = 1 & \forall j \in J \\
& z_i^A \ge \sum_{j\in J} f_j^A y_{ij}^A - p_i & \forall i \in I \\
& z_i^B \ge \sum_{j\in J} f_j^B y_{ij}^B - p_i & \forall i \in I \\
& z_i^C \ge \sum_{j\in J} f_j^C y_{ij}^C - p_i & \forall i \in I \\
& z_i^A, z_i^B, z_i^C \ge 0,\ \ x_i \in \{0,1\} & \forall i \in I \\
& y_{ij}^A, y_{ij}^B, y_{ij}^C \in \{0,1\} & \forall i \in I, j \in J
\end{array}
$$

Stochastic model II

Case II: in another problem, y_{ij} may correspond to a road from i to j (i.e. to be decided now)

$$
\begin{array}{lll}
\min & \sum_{i\in I} c_i x_i + \sum_{i\in I}\sum_{j\in J} d_{ij} y_{ij} + & \\
& +p^A(g\sum_{i\in I} z_i^A) + & \\
& +p^B(g\sum_{i\in I} z_i^B) + & \\
& +p^C(g\sum_{i\in I} z_i^C) & \\
& y_{ij} \le x_i & \forall i \in I, \forall j \in J \\
& \sum_{i\in I} y_{ij} = 1 & \forall j \in J \\
& z_i^A \ge \sum_{j\in J} f_j^A y_{ij} - p_i & \forall i \in I \\
& z_i^B \ge \sum_{j\in J} f_j^B y_{ij} - p_i & \forall i \in I \\
& z_i^C \ge \sum_{j\in J} f_j^C y_{ij} - p_i & \forall i \in I \\
& z_i^A, z_i^B, z_i^C \ge 0,\ \ x_i \in \{0,1\} & \forall i \in I \\
& y_{ij} \in \{0,1\} & \forall i \in I, j \in J
\end{array}
$$

ISE 426
Optimization models and applications

Lecture 22 — November 20, 2014

- Nonlinear Programming (NLP)
- Least squares example
- Quadratic Programming

Reading:

- Winston&Venkataramanan, Ch. 12 up to §12.4; §12.10

Nonlinear Programming (**NLP**)

Consider the **continuous** case: no integer variables.

$$\begin{aligned} \min \quad & f(x) \\ & g_i(x) \le 0 \quad \forall i = 1, 2 \dots, m \\ & h_j(x) = 0 \quad \forall j = 1, 2 \dots, p \\ & x \in \mathbb{R}^n \end{aligned}$$

This is convex when:

- $f(x)$ is convex
- all $g_i(x)$ are convex
- all $h_i(x)$ are convex and their opposite $-h(x)$ is convex too

$\Leftrightarrow$ they are linear (aka affine):

$$h_i(x) = a_i^\top x - b_i$$

Unconstrained Optimization Problem

What is the main good news about convex optimization?

$$\min f(x) \quad x \in R^n$$

If f is differentiable, then optimality condition:

可微 $\nabla f(x) = 0$

If f is convex, then this condition is sufficient.

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1}(x) \\ \vdots \\ \frac{\partial f(x)}{\partial x_n}(x) \end{bmatrix}$$

gradient. 梯度.

一阶导.

Jacobi 的特殊情形.

Optimization Problem

partial deviation
+ $h_j(x) = 0$
$\Rightarrow (x, \dots, \nu_j)$.

$$\begin{aligned} & \min f(x) \\ & g_i(x) \le 0, \quad i = 1, \dots, m \\ & h_j(x) = 0, \quad j = 1, \dots, p \end{aligned}$$

Lagrangian function

$L'(\nu_j)$

$$L(x, \lambda, \nu) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} \nu_j h_j(x)$$

Dual function

$$g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) \le \text{下界}$$

(λ, ν) is called dual feasible if $\lambda \ge 0$ and $g(\lambda, \nu) > -\infty$.

$\because g_i(x) \le 0$?

Dual Problem

Lagrangian function

$$L(x, \lambda, \nu) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} \nu_j h_j(x)$$

Dual function

$$d(\lambda, \nu) = \inf_x L(x, \lambda, \nu)$$

Want to find (λ, ν) such that $\text{argmin}_x L(x, \lambda, \nu)$ is the solution to the original problem.
Lower bound: $d(\lambda, \nu) \le f(x)$ for any feasible (λ, ν) and x.
Dual problem

$$\begin{aligned} \max \quad & d(\lambda, \nu) \\ & \lambda \ge 0 \end{aligned}$$

Weak and Strong Duality
Let f^* be solution of the primal problem and d^* - solution of the dual problem.

- Weak duality: $d^* \le f^*$, always.
- Strong duality: If primal is convex then (usually) $d^* = f^*$.

Optimality conditions

Lagrangian function

$$L(x, \lambda, \nu) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} \nu_j h_j(x)$$

KKT (Karush-Kuhn-Tucker) optimality conditions: assume f_i g_i and h_j are differentiable. If x^*, λ^*, ν^* are optimal with zero duality gap, then

$$g_i(x^*) \leq 0,\ h_i(x^*) = 0$$

$$\lambda_i \geq 0$$

$$\nabla f(x^*) + \sum_i \lambda_i^* \nabla g_i(x^*) + \sum_i \nu_i^* \nabla h_i(x) * = 0$$

$$\lambda_i^* g_i(x^*) = 0.$$

For convex problems: if x^*, λ^*, ν^* satisfy KKT conditions above, then they are optimal.

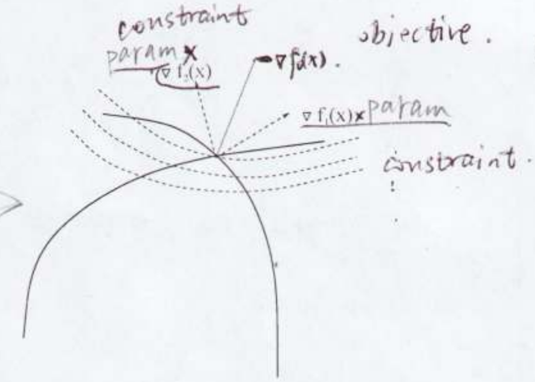
$\because g_i(x^*) \leq 0.$

① $g_i(x^*) = 0$. use this constraint

② $g_i(x^*) < 0$, $\lambda_i^* = 0$, slackness. this constraint is useless.

图形表达

Optimalty conditions



(Mixed-Integer) Nonlinear Programming (**MINLP**)

In the general form,

$$\begin{aligned} \min\ & f(x) \\ & g_i(x) \leq 0 \quad \forall i = 1, 2 \ldots, m \\ & h_j(x) = 0 \quad \forall j = 1, 2 \ldots, p \\ & x_k \in \mathbb{Z} \quad \forall k \in K \subseteq \{1, 2 \ldots, n\} \end{aligned}$$

A "convex" MINLP problem is such that relaxing integrality of all variables yields a convex (continuous) problem.

- Not really a convex problem, just an abuse of notation
- (integrality constraints make the problem nonconvex)

$\Rightarrow$ difficult, have to enumerate all local minima 列举.

- However, relaxing integrality gives a convex problem

$\Rightarrow$ it is relatively easy to find a lower bound

Example: solve the following problem by Branch&Bound:

$$\begin{aligned} \min\ & (x - \tfrac{1}{3})^2 \\ & x \in \mathbb{Z} \end{aligned}$$

(Mixed-Integer) Nonlinear Programming (**MINLP**)

In the general form,

$$\begin{aligned} \min\ & f(x) \\ & g_i(x) \leq 0 \quad \forall i = 1, 2 \ldots, m \\ & h_j(x) = 0 \quad \forall j = 1, 2 \ldots, p \\ & x_k \in \mathbb{Z} \quad \forall k \in K \subseteq \{1, 2 \ldots, n\} \end{aligned}$$

A "nonconvex" NLP can model any problems considered so far including integer variables. How?
In theory NLP=MINLP, but in practice MINLPs are the hardest to solve.

- The type of nonconvexity is unknown and hard to explore.

$\Rightarrow$ do not know where local minima are.

- Relaxing integrality does not give a convex problem.

Example: solve the following problem by Branch&Bound:

$$\begin{aligned} \min\ & -(\tfrac{1}{3}x^3 + \tfrac{1}{4}x^2 - \tfrac{1}{2}x) \\ & x \in \mathbb{Z},\ x \in [-10, 10] \end{aligned}$$

Equality constrained convex quadratic programming

$$\begin{aligned} \min\ & x^\top Q x + cx \\ & a_1 x = b_1 \\ & a_2 x = b_2 \\ & \vdots \\ & a_m x = b_m \end{aligned}$$

where Q is a square psd matrix.

~~PDS~~ positive semi-definite.

Positive (Semi)Definite Matrices

A square $n \times n$ matrix A is Positive Definite (PD) (denoted with $A \succ 0$) if, for any n-vector $x \neq 0$, the following holds:

$$x^\top A x \geq 0$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j =$$

$$\begin{array}{llll} a_{11}x_1^2 & +a_{12}x_1x_2 & + \ldots & +a_{1n}x_1x_n + \\ a_{21}x_2x_1 & +a_{22}x_2^2 & + \ldots & +a_{2n}x_2x_n + \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}x_nx_1 & +a_{n2}x_nx_1 & + \ldots & +a_{nn}x_n^2 \end{array}$$

$$> 0$$

最小两乘

Positive (Semi)Definite Matrices

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) (denote it with $A \succeq 0$) if all principal minors of A are nonnegative.

A symmetric matrix $A \in R^{n\times n}$ is **Positive Definite** (PD) (denote it with $A \succ 0$) if all leading principal minors of A are nonnegative.

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) if and only if $A = BB^\top$ for some $B \in R^{n\times n}$.

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) if for all $i = 1, \ldots, n$ $a_{ii} \geq \sum_{j=1, j\neq i}^{n} |a_{ij}|$.

A symmetric matrix $A \in R^{n\times n}$ is **Positive Semidefinite** (PSD) if all its eigenvalues are nonnegative.

Application: Least squares approximation

Problem: Perform regression analysis on a set of experimental observations to identify a trend.

- A set of $(n+1)$-dimensional points

$$\begin{aligned} a_1 &= (a_{11}, a_{12} \ldots, a_{1n}, b_1),\\ a_2 &= (a_{21}, a_{22} \ldots, a_{2n}, b_2),\\ &\vdots\\ a_m &= (a_{m1}, a_{m2} \ldots, a_{mn}, b_m),\end{aligned}$$

  are the result of the experiments
- i.e., each of them corresponds to a single observation and is a vector of numbers
- for instance, for each patient in a hospital, it is the vector of parameters: blood pressure, levels of cholesterol, etc.)

$a = \dfrac{\sum xy - \frac{1}{N}\sum x \sum y}{\sum x^2 - \frac{1}{N}(\sum x)^2}$, $b = \bar{y} - a\bar{x}$.

$y = ax + b.$

Application: Least squares approximation

- we think that these data are not random, but are connected to one another by a linear function
- i.e., there is a vector $(p_1, p_2 \ldots, p_n)$ and a scalar q such that

$$\begin{array}{lllllll} b_1 = & p_1a_{11} & +p_2a_{12} & +\ldots & +p_na_{1n} & +q\\ b_2 = & p_1a_{21} & +p_2a_{22} & +\ldots & +p_na_{2n} & +q\\ \vdots & \vdots & \vdots & & \ddots\ \vdots & \\ b_m = & p_1a_{m1} & +p_2a_{m2} & +\ldots & +p_na_{mn} & +q \end{array}$$

- ...but we don't know $p_1, p_2 \ldots, p_n$ or q.

Application: Least squares approximation

We may just solve the linear system in the previous slide, but:

- there may be errors in the observation
- to add some robustness to the observation, usually $m \gg n$

Thus, if there are errors or noise in the observation, we look for a vector p and scalar q that minimizes the the total error.

- Variables:
  - $p_i, i = 1, 2 \ldots, n$
  - q
- Objective:

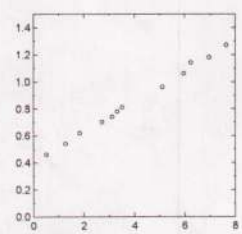
$$\sum_{i=1}^{m}(b_i - p_1a_{i1} - p_2a_{i2} \ldots - p_na_{in} - q)^2 = \sum_{i=1}^{m}(b_i - (\sum_{j=1}^{n} p_ja_{ij} + q))^2$$

$\Rightarrow$ a continuous (unconstrained) NLP problem.

Example

a_1	a_2	a_3	a_4	a_5	a_6
0.52	1.28	1.84	2.72	3.12	3.32
0.46	0.54	0.62	0.70	0.74	0.78

a_7	a_8	a_9	a_{10}	a_{11}	a_{12}
3.52	5.12	5.96	6.24	6.96	7.64
0.81	0.96	1.06	1.14	1.18	1.27



Are there p_1 and q satisfying:

$$\begin{aligned} 0.46 &= 0.52p_1 + q\\ 0.54 &= 1.28p_1 + q\\ &\vdots\\ 1.27 &= 7.64p_1 + q \end{aligned}$$

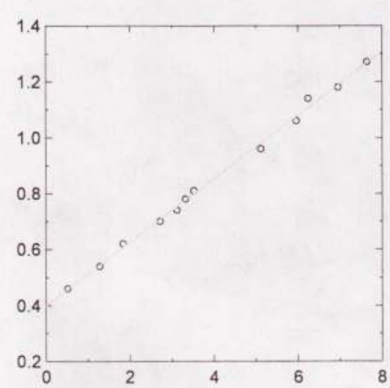
(I don't think so...)

At least minimize total error:

$$\begin{aligned} \min\ & (0.46 - 0.52p_1 - q)^2 +\\ +\ & (0.54 - 1.28p_1 - q)^2\\ +\ & \vdots\\ +\ & (1.27 - 7.64p_1 - q)^2 \end{aligned}$$

Result

$p = 0.113495$, $q = 0.39875$.



Least squares - convex quadratic problem

We solve

$$
\begin{aligned}
\min \quad & \tfrac{1}{2}\sum(a_i^\top x - b_i)^2 = \\
& \cancel{\tfrac{1}{2}\|Ax - b\|^2 =} \\
& \tfrac{1}{2}(Ax - b)^\top(Ax - b) = \\
& \tfrac{1}{2}x^\top A^\top Ax - b^\top Ax + \tfrac{1}{2}b^\top b \\
& \tfrac{1}{2}x^\top Qx + c^\top x + const = \\
& f(x)
\end{aligned}
$$

Q is positive semidefinite because $Q = A^\top A$.
Convex unconstrained problem $\Rightarrow$ every local minimum is a global minimum! To solve set derivative to zero:

$$\nabla f(x) = Qx + c = 0$$

$$x = -Q^{-1}c \quad \text{if } Q \succ 0 \; - \; \text{One solution!}$$

If Q is singular - two cases:

1. Unbounded solution
2. Infinite number of solutions

Another example of convex quadratic problem

You are considering purchasing a stock over the next n days. You can buy it on any number of the days. Since you represent a large investor and purchase large amount of the stock each time, your purchases affect the price of the stock. If you purchase the y_i amount of the stock on day i then the price on that day goes up by θy_i on that day, by next day the effect of your purchase is dampened a bit and the price raised by $\frac{\theta}{2}y_i$, then the next day it is $\frac{\theta}{4}y_i$ and so on. If the starting price on day i is p_i derive the optimal purchasing schedule over the next 10 days is you need to purchase the total of Y.

Formulation via a convex QP

- Variables y_i - the amount of stock purchased on day i.
- The price of the stock on day 1 is $\theta y_1 + p_1$.
- The price of the stock on day 2 is $\frac{\theta}{2}y_1 + \theta y_2 + p_2$.
- .....
- The price of the stock on day i is

$$\frac{\theta}{2^{i-1}}y_1 + \frac{\theta}{2^{i-2}}y_2 + \ldots + \theta y_i + p_i.$$

We need to solve

$$
\begin{aligned}
\min \quad & \textstyle\sum_{i=1}^n y_i\left(\sum_{j=1}^i \frac{\theta}{2^{i-j}}y_j + p_i\right) = \\
& \tfrac{1}{2}y^\top Qy + p^\top y \\
\text{s.t.} \quad & \textstyle\sum_{i=1}^n y_i = Y
\end{aligned}
$$

要求是对称矩阵.

Formulation via a convex QP

$$
\begin{aligned}
\min \quad & \tfrac{1}{2}y^\top Qy + p^\top y \\
\text{s.t.} \quad & \textstyle\sum_{i=1}^n y_i = Y
\end{aligned}
$$

y_{ii}

$\frac{1}{2}\times$ (多存一次)

$$Q = \begin{pmatrix} 2\theta & \frac{\theta}{2} & \cdots & \frac{\theta}{2^{n-1}} \\ \frac{\theta}{2} & 2\theta & \cdots & \frac{\theta}{2^{n-2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\theta}{2^{n-1}} & \frac{\theta}{2^{n-2}} & \cdots & 2\theta \end{pmatrix}$$

Q is positive definite because $Q_{ii} > \sum_{j\neq i}|Q_{ij}|$.
Convex quadratic problem $\Rightarrow$ every local minimum is a global minimum!
Only equality constraint - can solve by solving a system of linear equations:

$Qx + \lambda + p = 0$

$$\begin{bmatrix} Q & e \\ e^\top & 0 \end{bmatrix}\begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} -p \\ Y \end{pmatrix}$$

$e^\top x = \Sigma x = Y$

If the matrix of the system is singular - two cases:

1. Unbounded solution

但是

Lagrangian function.

Conex Quadratic Programming (QP)

$$
\begin{aligned}
\min \quad & x^\top Qx + cx \\
& a_1x \geq b_1 \\
& a_2x \geq b_2 \\
& \vdots \\
& a_mx \geq b_m
\end{aligned}
$$

where Q is a square psd matrix.

Application: Markowitz porfolio selection

- n stocks
- Decision variables $x_j, j = 1, \ldots, n$ represent number of shares of stock j to purchase.
- μ_j is mean of return of stock j (say after 1 year).
- σ_{jj} is variance of return of stock j (measures risk).
- σ_{ij} $(i \neq j)$ is covariance of return on one share of stock i and one of stock j.
- These parameters must all be estimated in practice.

Markowitz model

$$
\begin{aligned}
\min \quad & V(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_{ij} x_i x_j \\
\text{s.t.} \quad & \sum_{j=1}^{n} \mu_j x_j \geq L \\
& \sum_{j=1}^{n} P_j x_j \leq B \\
& x_j \geq 0 \quad \forall j = 1, \ldots, n
\end{aligned}
$$

- L is minimum acceptable expected return.
- P_j is price per share of stock j (today).
- B is budget.

Robust Markowitz model

$$
\begin{aligned}
\min \quad & V(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_{ij} x_i x_j \\
\text{s.t.} \quad & \sum_{j=1}^{n} \mu_j x_j \geq L \quad \mu_i \text{ is not known} \\
& \sum_{j=1}^{n} P_j x_j \leq B \\
& x_j \geq 0 \quad \forall j = 1, \ldots, n
\end{aligned}
$$

- L is minimum acceptable expected return.
- μ is not known, but has to be in some set M.
- How do we solve this?

Robust Markowitz model

$$
\begin{aligned}
\min \quad & V(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_{ij} x_i x_j \\
\text{s.t.} \quad & \min_{\mu \in M} \sum_{j=1}^{n} \mu_j x_j \geq L \\
& \sum_{j=1}^{n} P_j x_j \leq B \\
& x_j \geq 0 \quad \forall j = 1, \ldots, n
\end{aligned}
$$

- Optimize risk subject to minimum return requirement in the worst case.
- M can be a cubic set: $\{\mu : \mu_i^1 \leq \mu_i \leq \mu_i^2, \forall i = 1, \ldots, n\}$.
- M can be a ball: $\{\mu : \|\mu - \mu_0\| \leq R\}$.

2范

MINLP Markowitz model

Say a portfolio can only have k stocks.

$$
\begin{aligned}
\min \quad & V(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_{ij} x_i x_j \\
\text{s.t.} \quad & \sum_{j=1}^{n} \mu_j x_j \geq L \\
& \sum_{j=1}^{n} P_j x_j \leq B \\
& x_i \leq B y_i \\
& \sum y_i \leq k \\
& x_j \geq 0,\ y_i \in \{0, 1\} \quad \forall j = 1, \ldots, n
\end{aligned}
$$

考虑组合个数.

A convex MINLP commonly used in financial models.

Convex quadratically constrained quadratic problems (QCQP)

QCQPs are problems of the form

$$
\begin{aligned}
\min \quad & x^\top Q^0 x + a_0 x \\
& x^\top Q^1 x + a_1 x \leq b_1 \\
& x^\top Q^2 x + a_2 x \leq b_2 \\
& \vdots \\
& x^\top Q^m x + a_m x \leq b_m
\end{aligned}
$$

where $Q_0, Q_1 \ldots, Q_m$ are square psd matrices.

Convex quadratically constrained quadratic problems (QCQP)

QCQPs are problems of the form

$$
\begin{aligned}
\min \quad & \textstyle\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}^0 x_i x_j + \sum_{j=1}^{n} a_{0j} x_j \\
& \textstyle\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}^1 x_i x_j + \sum_{j=1}^{n} a_{1j} x_j \leq b_1 \\
& \textstyle\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}^2 x_i x_j + \sum_{j=1}^{n} a_{2j} x_j \leq b_2 \\
& \vdots \\
& \textstyle\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}^m x_i x_j + \sum_{j=1}^{n} a_{mj} x_j \leq b_m
\end{aligned}
$$

where $Q_0, Q_1 \ldots, Q_m$ are square psd matrices.

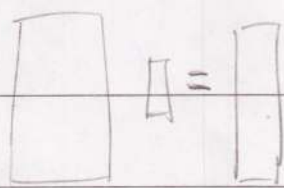
ISE426 Fall 2015

Lecture 23 – November 24, 2015

Convex quadratic programming in Machine Learning:

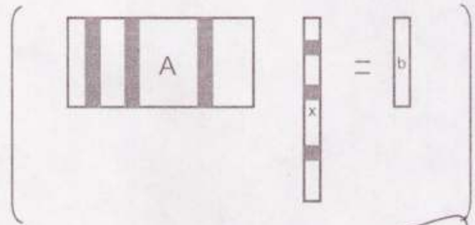
Sparse Optimization

Support Vector Machines and



$: ||Ax-b||^2$

Lasso (sparse LS regression)



$Ax \approx b$

$\boxed{x \neq 0}$:

x has few nonzero elements: $\|x\|_0$ is small!

$||Ax-b||^2 - \rho||x||^2$ $\qquad \|x\|_0 = \text{length}(b)$.

Example from gene expression

- Single Nucleotide Polymorphism (SNP) – point sites of variation in traits
- Each SNP associated with two alleles (states)

Classifying state of a disease based on some of the SNPs

Not known which SNPs are important – use feature selection

600,000 SNPs and 5,000 individuals/data points.

feature selection

Sparse solutions

$y_i = \begin{cases} 1, & \text{if } x \neq 0 \\ 0, & \text{if } x = 0 \end{cases}$

Sparse signal reconstruction

$$\begin{aligned} \min \quad & \|x\|_0 \\ s.t. \quad & Ax = b. \end{aligned}$$

x的0次方和 $\quad \min \sum y_i$

Sparse solution $x \in \mathbf{R}^n$, matrix $A \in \mathbf{R}^{m\times n}, n >> m$

The system is underdetermined, but if card(x)<m, can recover signal.

How do we formulate this as an MILP?

$$\begin{aligned} \min \quad & \sum y_i \\ s.t. \quad & Ax = b. \\ & x_i \le My_i,\ i = 1,\dots,n \\ & -x_i \le My_i\ i = 1,\dots,n \\ & y_i \in \{0,1\}\ i = 1,\dots,n \end{aligned}$$

Sparse solution using l_1-norm

The problem is difficult in general. Typical relaxation,

$$\begin{aligned} \min \quad & \sum y_i \\ s.t. \quad & Ax = b. \\ & x_i \le My_i,\ i = 1,\dots,n \\ & -x_i \le My_i\ i = 1,\dots,n \\ & 0 \le y_i \le 1\ i = 1,\dots,n \end{aligned}$$

relax to $y_i \in [0,1]$

$$\Downarrow$$

$$\begin{aligned} \min \quad & \sum \frac{|x_i|}{M} \\ s.t. \quad & Ax = b. \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} \min \quad & \|x\|_1 \\ s.t. \quad & Ax = b. \end{aligned}$$

$\min \sum |x_i|$, $\sum|x_i| = ||x||_1$

$||x||_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$

Sparse solutions using the l_1-norm

Sparse signal reconstruction

$$\begin{aligned} \min \quad & \|Ax - b\| \\ s.t. \quad & \|x\|_0 \le k \end{aligned}$$

k-sparse signal $x \in \mathbf{R}^n$, matrix $A \in \mathbf{R}^{m\times n}, n >> m$

The system is underdetermined, but if card(x)<k, can recover signal.

How do we formulate this as an MILP?

$$\begin{aligned} \min \quad & \|Ax - b\|^2 \\ s.t. \quad & \sum y_i \le k \\ & x_i \le My_i,\ i = 1,\dots,n \\ & -x_i \le My_i\ i = 1,\dots,n \\ & y_i \in \{0,1\}\ i = 1,\dots,n \end{aligned}$$

Recovery by using the l_1-norm

The problem is difficult in general. Typical relaxation,

$$\begin{aligned} \min \quad & \|Ax-b\| \\ s.t. \quad & \sum y_i \le k \\ & x_i \le My_i,\ i=1,\ldots,n \\ & -x_i \le My_i\ i=1,\ldots,n \\ & 0 \le y_i \le 1\ i=1,\ldots,n \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} \min \quad & \|Ax-b\|^2 \\ s.t. \quad & \sum \frac{|x_i|}{M} \le k \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} \min \quad & \|Ax-b\|^2 \\ s.t. \quad & ||x||_1 \le t(=kM?) \end{aligned}$$

Other formulations

Regularized regression or Lasso:

$$\min \quad \frac{1}{2}||Ax-b||^2 + \lambda||x||_1$$

Sparse regressor selection

Noisy signal recovery

$$\begin{aligned} \min \quad & ||Ax-b|| \\ s.t. \quad & ||x||_1 \le t. \end{aligned}$$

$$\begin{aligned} \min \quad & ||x||_1 \\ s.t. \quad & ||Ax-b|| \le \epsilon. \end{aligned}$$

Types of convex problems

$$\min \quad \frac{1}{2}||Ax-b||^2 + \lambda||x||_1$$

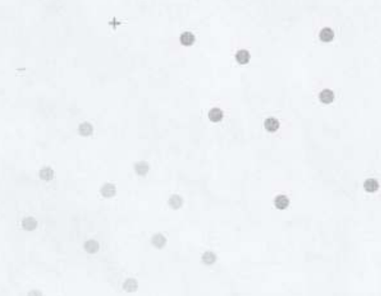
Variable substitution: $x = x' - x'',\ x' \ge 0,\ x'' \ge 0$

$$\begin{aligned} \min \quad & \frac{1}{2}||A(x'-x'')-b||^2 + \lambda(x'+x'') \\ \text{s.t.} \quad & x' \ge 0, x'' \ge 0 \end{aligned}$$

Convex non-smooth objective with linear inequality constraints

Binary classification problem

Two sets of labeled points

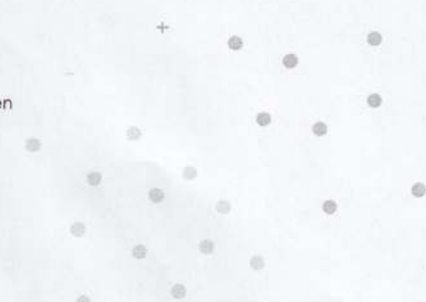


Binary classification problem

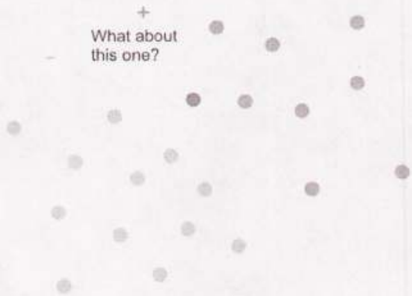
How to label this new point?

Binary classification problem

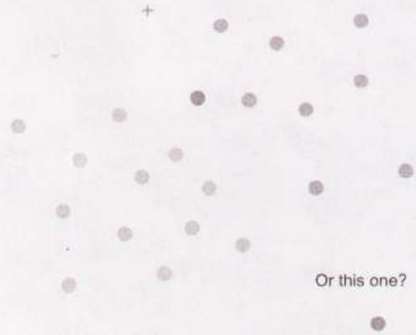
Probably green



Binary classification problem

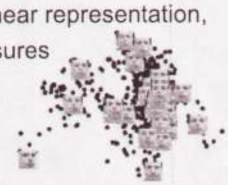


Binary classification problem



Examples from image classification

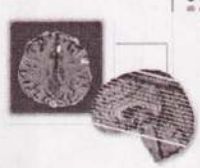
- Optical character recognition
  - Automatically read digits in zip code
    - 256 dim vector of pixels, 10 classes,
    - classification or clustering task
- Face recognition and detection
  - much larger dimension, nonlinear representation,
  - Non-euclidean similarity measures



Examples from text and internet

- Text categorization
  - detect spam/nonspam emails
    - Many possible features
    - False positives are very bad, false negatives are OK.
    - Online setting possible, huge data sets.
  - choose articles of interest to individualize news sites
    - Large dimension – size of dictionary, small training set, possibly online setting
    - Only few words are important.
- Ranking
  - Predict a page rank for a given a search query
    - How to do it? Predict relative ranks of each pair of pages?

Examples from Medicine



- Functional Magnetic resonance imaging
  - Uses a standard MRI scanner to acquire functionally meaningful brain activity
  - Measures changes in blood oxygenation
  - Non-invasive, no ionizing radiation
  - Good combination of spatial / temporal resolution
    - Voxel sizes ~4mm
    - Time of Repetition (TR) ~1s

    About 30000 voxels are active and measured.
  - Only a few (probably) contribute to what the subject is "feeling" during the experiment (anger, frustration, boredom..)
- Breast cancer risk patients
  - Take several measurements of a patient and some basic characteristics an predict if the patient is at high risk
  - Low dimensional, but very different attributes. Large scale data.
  - May involve "active learning" – additional labels obtained by involving more tests or a professional.
  - KDD 2008 cup challenge

fMRI image courtesy of fMRI Research Center © Columbia University

The binary classification problem

- The universe of data-label pairs (x, y),
- $y \in \{+1, -1\}$ for all $x \in \mathbf{R}^m$.
- Given a set $X \subset \mathbf{R}^m$ of n vectors.
- For each $x_i \in X$ the label y_i is known.
- Find a function $f(x) \approx y$

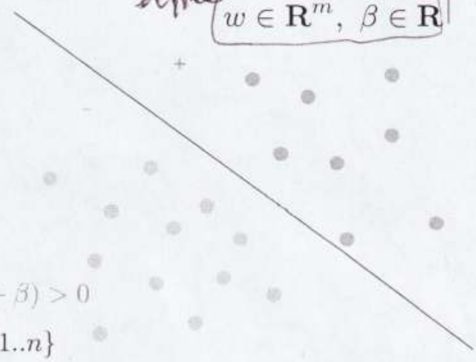
Linear classifier

Idea: separate a space into two half-spaces

Linear classifier

line $\boxed{\begin{array}{c} w^\top x + \beta = 0 \\ w \in \mathbf{R}^m,\ \beta \in \mathbf{R} \end{array}}$

Like this:



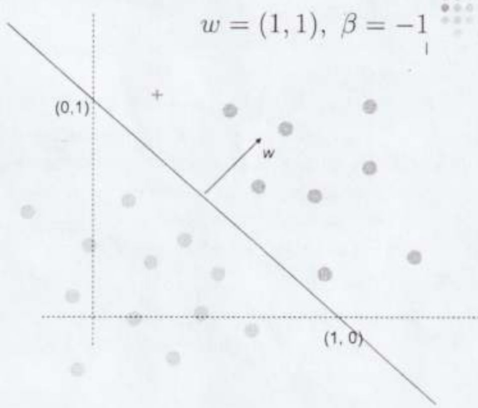
$y_i(w^\top x_i + \beta) > 0$

$\forall i \in \{1..n\}$

Linear classifier

$x_1 + x_2 - 1 = 0$

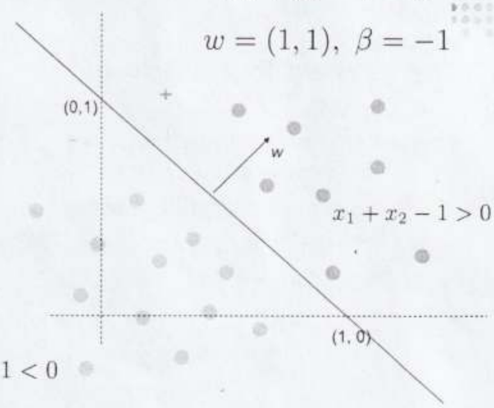
$w = (1,1),\ \beta = -1$



Linear classifier

$x_1 + x_2 - 1 = 0$

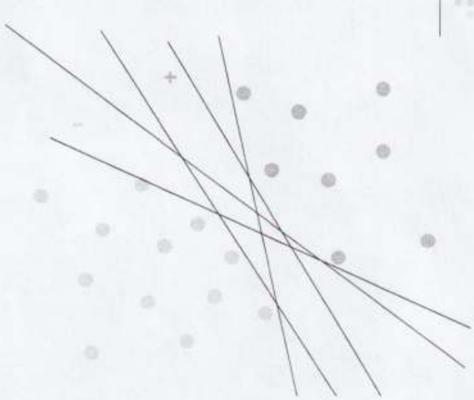
$w = (1,1),\ \beta = -1$



$x_1 + x_2 - 1 > 0$

$x_1 + x_2 - 1 < 0$

Linear classifier

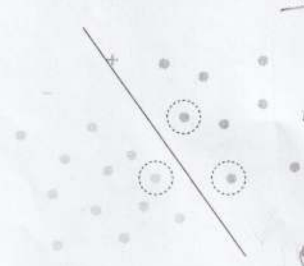


Support vector machines

结果标签.

Assume each x_i is not known exactly,
but $z_i \in B(x_i, r)$

$y_i = \pm 1$. can do this.



$$\min_{z_i \in B_i} y_i(w^\top z_i + \beta) \geq 0,\ \forall i \in \{1..n\}$$

$$\Downarrow$$

$$y_i(w^\top x_i + \beta) - \frac{r}{\|w\|} w^\top w \geq 0,\ \forall i \in \{1..n\}$$

$$\Downarrow$$

$$y_i(w^\top x_i + \beta) - \|w\| r \geq 0,\ \forall i \in \{1..n\}$$

函数距离 几何距离

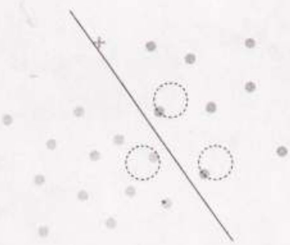
Find the largest r or the smallest $||w||$

这里令函数距离为1.

Support vector machines

Assume each x_i is not known exactly, but $z_i \in B(x_i, r)$

$||w|| = \sqrt{w_1^2 + \dots + w_n^2}$



$$\min_{z_i \in B_i} y_i(w^\top z_i + \beta) \geq 0,\ \forall i \in \{1..n\}$$

$$\Downarrow$$

$$y_i(w^\top x_i + \beta) - \frac{r}{\|w\|} w^\top w \geq 0,\ \forall i \in \{1..n\}$$

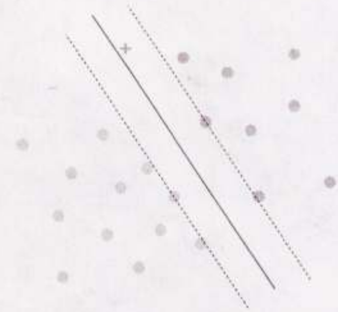
$$\Downarrow$$

$$y_i(w^\top x_i + \beta) - \|w\| r \geq 0,\ \forall i \in \{1..n\}$$

Find the largest r or the smallest $||w||$

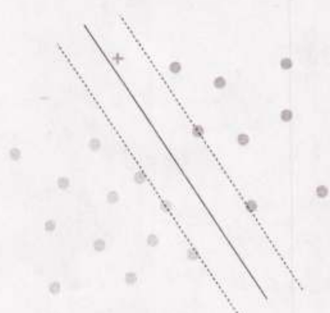
Support vector machines

$$\min_{w,\beta} ||w||,\ \text{s.t. } y_i(w^\top x_i + \beta) - 1 \geq 0,\ \forall i \in \{1..n\}$$



Support vector machines

$$\min_{w,\beta} \frac{1}{2}||w||^2,\ \text{s.t. } y_i(w^\top x_i + \beta) - 1 \geq 0,\ \forall i \in \{1..n\}$$



Optimization Problem

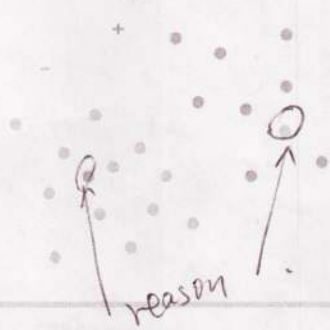
Total number of data points: n

$$\begin{aligned} \min_{w \in \mathbf{R}^m, \beta \in \mathbf{R}} \quad & \frac{1}{2} w^\top w \\ \text{s.t.} \quad & y_i(w^\top x_i + \beta) \geq 1, \quad i = 1, \dots, n \end{aligned}$$

How many variables? Constraints? What can go wrong?

Support vector machines

$y_i(w^\top x_i - b) - 1 \geq 0,\ \forall i \in \{1..n\}$ – no such w!



Soft margin SVM

Total number of data points: n

$$\begin{aligned} \min_{\xi, w, \beta} \quad & \frac{1}{2} w^\top w \\ \text{s.t.} \quad & y_i(w^\top x_i + \beta) \geq 1 - \xi_i, \quad i = 1, \dots, n \end{aligned}$$

What's wrong with this formulation?

Soft margin SVM

Total number of data points: n

penalty

$$\min_{\xi,w,\beta} \quad \frac{1}{2}w^\top w + c\sum_{i=1}^{n}\xi_i$$
$$\text{s.t.} \quad y_i(w^\top x_i + \beta) \geq 1-\xi_i, \quad i=1,\ldots,n$$
$$\xi \geq 0, \qquad i=1,\ldots,n.$$

How many variables? Constraints?

Soft margin SVM

Total number of data points: n

$$\min_{\xi,w,\beta} \quad \frac{1}{2}w^\top w + c\sum_{i=1}^{n}\xi_i$$
$$\text{s.t.} \quad y_i(w^\top x_i + \beta) \geq 1-\xi_i, \quad i=1,\ldots,n$$
$$\xi \geq 0, \qquad i=1,\ldots,n.$$

How many variables? Constraints?

What if n is very large? What if m is very large?

Optimization Problem

$$\min_{\xi,w,\beta} \quad \frac{1}{2}w^\top w + c\sum_{i=1}^{n}\xi_i$$
$$\text{s.t.} \quad y_i(w^\top x_i + \beta) \geq 1-\xi_i, \quad i=1,\ldots,n$$
$$\xi \geq 0, \qquad i=1,\ldots,n.$$

Every optimization problem has:

1. optimality conditions and 2. dual problem

Optimization Problem

$$\boxed{\text{At optimality } w^* = \textstyle\sum_{i=1}^{n}\alpha_i y_i x_i, \quad 0 \leq \alpha_i \leq c}$$

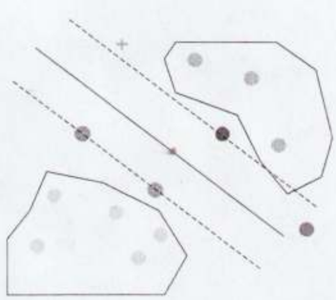
$$||w^*||^2 = (\sum_{i=1}^{n}\alpha_i y_i x_i)^\top(\sum_{i=1}^{n}\alpha_i y_i x_i) = \sum_{i,j=1}^{n} y_i y_j x_i^\top x_j \alpha_i \alpha_j$$

$$\min_{\alpha,\beta,\xi} \quad \frac{1}{2}\sum_{i,j=1}^{n} y_i y_j x_i^\top x_j \alpha_i \alpha_j + c\sum_{i=1}^{n}\xi_i$$
$$\text{s.t.} \quad \sum_j y_i y_j x_i^\top x_j \alpha_j + y\beta + \xi_i \geq 1, \quad i=1,\ldots,n$$
$$\xi_i \geq 0,\ 0 \leq \alpha_i \leq c, \qquad i=1,\ldots,n,$$

How many variables? Constraints?

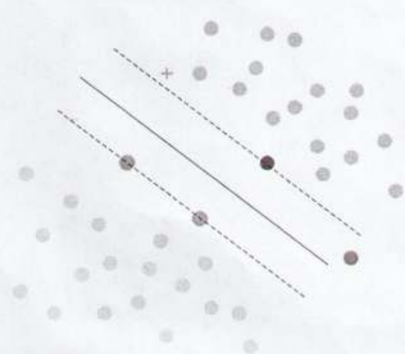
Support Vectors

- $0<\alpha<c$, $\xi=0$
- $\alpha=0$, $\xi=0$
- $\alpha=c$, $\xi>0$

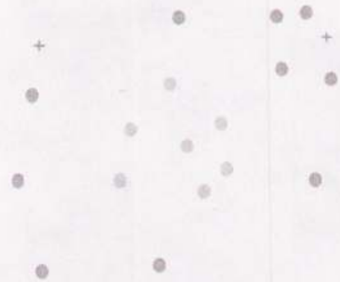


Support Vectors

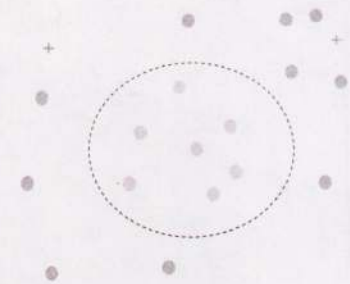
- $0<\alpha<c$, $\xi=0$
- $\alpha=0$, $\xi=0$
- $\alpha=c$, $\xi>0$



Oh, no! What do we do now?



Kernel SVM

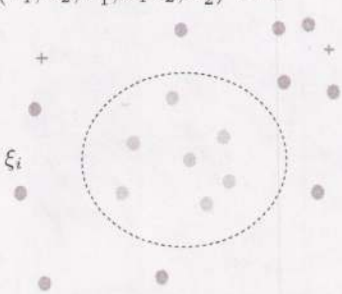


Kernel SVM

$$w_1x_1 + w_2x_2 + w_3x_1^2 + w_4x_1x_2 + w_5x_2^2 + \beta$$

$$w^\top \phi(x) + \beta, \ \phi(x) = (x_1, x_2, x_1^2, x_1x_2, x_2^2) \in \mathbf{R}^5$$

$$y_i(w^\top \phi(x_i) + \beta) \geq 1 - \xi_i$$



Optimization Problem

$$\boxed{\text{At optimality } w^* = \textstyle\sum_{i=1}^n \alpha_i y_i x_i, \quad 0 \leq \alpha_i \leq c}$$

$$||w||^2 = (\sum_{i=1}^n \alpha_i y_i x_i)^\top (\sum_{i=1}^n \alpha_i y_i x_i) = \sum_{i,j=1}^n y_i y_j x_i^\top x_j \alpha_i \alpha_j$$

$$\begin{aligned} \min_{\alpha,\beta,\xi} \quad & \frac{1}{2}\sum_{i,j=1}^n y_i y_j x_i^\top x_j \alpha_i \alpha_j + c\sum_{i=1}^n \xi_i \\ \text{s.t.} \quad & \sum_j y_i y_j x_i^\top x_j \alpha_j + y\beta + \xi_i \geq 1, \quad i = 1, \dots, n \\ & \xi_i \geq 0, \ 0 \leq \alpha_i \leq c, \qquad i = 1, \dots, n, \end{aligned}$$

Optimization Problem

$$\boxed{\text{At optimality } w^* = \textstyle\sum_{i=1}^n \alpha_i y_i \phi(x_i), \quad 0 \leq \alpha_i \leq c}$$

$$||w||^2 = (\sum_{i=1}^n \alpha_i y_i \phi(x_i))^\top (\sum_{i=1}^n \alpha_i y_i \phi(x_i)) = \sum_{i,j=1}^n y_i y_j \phi(x_i)^\top \phi(x_j) \alpha_i \alpha_j$$

$$\begin{aligned} \min_{\alpha,\beta,\xi} \quad & \frac{1}{2}\sum_{i,j=1}^n y_i y_j \phi(x_i)^\top \phi(x_j) \alpha_i \alpha_j + c\sum_{i=1}^n \xi_i \\ \text{s.t.} \quad & \sum_j y_i y_j \phi(x_i)^\top \phi(x_j) \alpha_j + y\beta + \xi_i \geq 1, \quad i = 1, \dots, n \\ & \xi_i \geq 0, \ 0 \leq \alpha_i \leq c, \qquad i = 1, \dots, n, \end{aligned}$$

How many variables? Constraints?

Kernel SVM

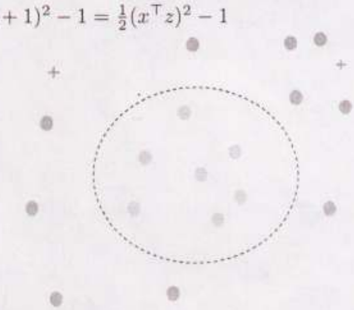
$$\phi(x) = (x_1, x_2, \tfrac{1}{\sqrt{2}}x_1^2, x_1x_2, \tfrac{1}{\sqrt{2}}x_2^2)$$

$$\phi(x)^\top \phi(z) = (x_1z_1 + x_2z_2 + \tfrac{1}{2}x_1^2z_1^2 + x_1x_2z_1z_2 + \tfrac{1}{2}x_2^2z_2^2)$$

O(m^2)

$$\phi(x)^\top \phi(z) = \tfrac{1}{2}(x_1z_1 + x_2z_2 + 1)^2 - 1 = \tfrac{1}{2}(x^\top z)^2 - 1$$

O(m)

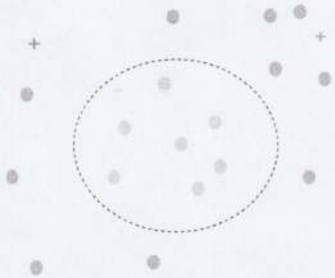


Kernel SVM

$Q_{ij} = y_i y_j {x_i}^\top x_j \rightarrow Q_{ij} = y_i y_j \phi(x_i)^\top \phi(x_j) = y_i y_j K(x_i, x_j)$
Kernel operation: $K(x_i, x_j) = \phi(x_i)^\top \phi(x_j)$
Examples:

- $K(x_i, x_j) = ({x_i}^\top x_j / a_1 + a_2)^d$
- $K(x_i, x_j) = \exp^{-||x_i - x_j||^2 / 2\sigma^2}$

$\phi(x) \in R^\infty$



Review.

1. The sum of convex function is a convex function.

2. principle minor $\boxed{all}$ $H_f(x) \geq 0$.

3. $g(x) = x_1^2 + x_2^2$ convex. $g(x) \leq 0$ convex. $g(x) \geq 0$ nonconvex

   $g(x) \geq b$ may be nonconvex.

4. $h(x)$ is linear $\Leftrightarrow \begin{cases} h(x) & \text{convex} \\ -h(x) & \text{convex} \end{cases}$

5. convex, concave. [illegible] ; nonconvex includes concave and ∼

6. convex optimization problems are easy:

   If a problem P is convex, a local optimum x^* of P is also a global optimum of P.

7. P: $\min f_0(x)$ st $f_1(x) \leq b_1$, $f_2(x) \leq b_2$, $\vdots$, $f_m(x) \leq b_m$

   set of points $F = \{ x \in \mathbb{R}^n : f_1(x) \leq b_1, f_2(x) \leq b_2, \ldots, f_m(x) \leq b_m \}$

   P': $\min \{ f_0'(x) : x \in F' \}$

   P' is a relaxation of $P \Leftrightarrow \begin{cases} F' \supseteq F \\ f_0'(x) \leq f_0(x) \text{ for all } x \in F \end{cases}$

8. lower bound $\leq Z_{optimal} \leq$ upper bound

9. In a knapsack problem, local optimum is changing the i's location without getting better

   using $\frac{P_i}{w_i}$ find lower bound, than try number larger than lower bound one by one.

10. Transportation model: 1). capacity: $\sum_{j=1}^{7} x_{ij} \leq P_i \ \forall i$; 2). demand: $\sum_{i=1}^{5} x_{ij} \geq d_j$. 3) $\underline{x_{ij} \geq 0}$ 35个

    to find a corner, need ≥ 35 个 equations. Thus, at least $(35 - 12 = 23)$ 个 $x_{ij} = 0$.

11. Production planning: 1). $x_i \leq P$, $\forall i = 1, 2, \ldots, 12$; 2). $y_i \leq C$, $\forall i = 1, 2, \ldots, 11$; 3). $y_0 = 0$, $y_{12} = 0$;

    4). $x_i + y_{i-1} = d_i + y_i$, $\forall i = 1, 2, \ldots, 12$.

12. The shortest path problem: $\min \sum_{(i,j) \in A} c_{ij} x_{ij}$

    $$\text{s.t. } \sum_{j \in V: (i,j) \in A} x_{ij} - \sum_{j \in V: (j,i) \in A} x_{ji} = b_i \quad \forall i \in V$$

    $x_{ij} \geq 0$. $\quad b_i = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise.} \end{cases}$

13. Min-Cost-Flow.

    $\min \sum_{(i,j) \in A} d_{ij} x_{ij}$

    s.t. $\sum_{j \in V: (j,i) \in A} x_{ji} = \sum_{j \in V: (i,j) \in A} x_{ij} \ \forall i \in V: s \neq i \neq t$

    $\sum_{j \in V: (s,j) \in A} x_{sj} = r$

    $0 \leq x_{ij} \leq c_{ij} \quad \forall (i,j) \in A$

Max-Flow:

$\max \sum_{j \in V: (j,t) \in A} x_{jt}$

s.t. $\sum_{j \in V: (j,i) \in A} x_{ji} = \sum_{j \in V: (i,j) \in A} x_{ij}$. $\quad \forall i \in V, s \neq i \neq t$.

$0 \leq x_{ij} \leq c_{ij} \quad \forall (i,j) \in A$.

Min-Cut: $\min \sum_{(i,j) \in A} c_{ij} z_{ij}$

st. $z_{ij} \geq \mu_j - \mu_i$

$\mu_s = 0$

$\mu_t = 1$

$0 \leq z_{ij} \quad \forall (i,j) \in A$

$0 \leq \mu_i \quad \forall i \in V$.

1. $\min\ c^T x$ $\qquad$ $\max\ b^T u$

$Ax \geq b$ $\qquad$ $A^T u \leq c$

$x \geq 0$ $\qquad$ $u \geq 0.$

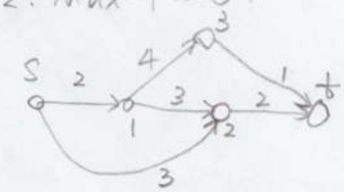
Minimize	Maximize
Var ≥ 0	S.t. $\leq$
Var ≤ 0	S.t. $\geq$
Var. Unrstrtd	S.t. $=$
S.t. $\leq$	Var ≤ 0
S.t. $\geq$	Var ≥ 0
S.t. $=$	Var. Unrstrtd.

$$c^T \bar{x} \geq (A^T \bar{u})^T \bar{x} = \bar{u}^T A \bar{x} \geq \underset{\overset{\shortparallel}{b^T \bar{u}.}}{\bar{u}^T b}$$

Let $c^T \bar{x} = b^T \bar{u}$,

$\begin{cases} (c^T - A^T \bar{u})\bar{x} = 0 \\ \bar{u}(A\bar{x} - b) = 0 \end{cases}$ complementary Slackness : $\begin{cases} \bar{u}_i(\sum_{i=1}^{n} a_{ji}\bar{x}_i - b_j) = 0 \\ \bar{x}_j(\sum_{j=1}^{m} a_{ji}\bar{u}_j - c_j) = 0. \end{cases}$

2. Max-Flow



primal :

$\max\ x_{2t} + x_{3t}$

$x_{s1} = x_{12} + x_{13}$

$x_{s2} + x_{12} = x_{2t}$

$x_{13} = x_{3t}$

$0 \leq x_{s1} \leq 2$ $\quad$ $0 \leq x_{s2} \leq 3$

$0 \leq x_{12} \leq 3$ $\quad$ $0 \leq x_{13} \leq 4$

$0 \leq x_{2t} \leq 2$ $\quad$ $0 \leq x_{3t} \leq 1.$

dual:

$\min\ 2z_{s1} + 3z_{12} + 2z_{2t} + \cdots + 4z_{13} + z_{3t}.$

$u_1 + z_{s1} \geq 0$

$u_2 + z_{s2} \geq 0$

$-u_1 + u_2 + z_{12} \geq 0$

$\vdots$

$-u_3 + z_{3t} \geq 1$

$z_{ij} \geq 0$, u_i - unrstrtd.

3. The shortest Path problem

primal : $\min\ c_{AB}x_{AB} + c_{BA}x_{BA} + \cdots + c_{EF}x_{EF} + c_{FE}x_{FE}$

(점A) $x_{AB} + x_{AC} - x_{BA} - x_{CA} = 1$

(점B) $x_{BA} + x_{BC} + x_{BD} - x_{AB} - x_{CB} - x_{DB} = 0$

$\vdots$

(점F) $x_{FB} + x_{FD} + x_{FE} - x_{BF} - x_{DF} - x_{EF} = -1$

$x_{AB}, x_{BA}, \cdots x_{FE} \geq 0.$

dual : $\max\ u_A - u_F$

$u_A - u_B \leq c_{AB}$

$u_B - u_A \leq c_{BA}$

$u_A - u_C \leq c_{AC}$

$\vdots$

$u_F - u_E \leq c_{FE}$

$u_A, u_B \cdots u_F$ - unstrtd

About Convex : ① Linear $\left|\begin{array}{l}\text{inequality} \\ \text{equality}\end{array}\right.$; ② sum of the convex functions

③. Positive Definite Hessian

④ NOT : Non-linear equality.

4. $f(x) = x^{2n}$ $\qquad$ $f(x) = e^x$, $f(x) = -\ln(x)$

1). min max { linear, ... } | max min { linear, ... }

$y \geq$ all linear | $y \leq$ all linear.

min y. | max y.

2). Job assignment : fair : $\max \min_{i=1,2,\cdots,m} \sum_{j=1}^{m} c_{ij} x_{ij}$.

$$\sum_{j=1}^{m} x_{ij} = 1 \quad , \forall i = 1, 2, \cdots, m$$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad , \forall j = 1, 2, \cdots, m.$$

$$x_{ij} \in \{0, 1\} \quad \forall i,j = 1, 2, \cdots, m$$

3). $\min \max_{k=1,2,\cdots H} \left| \sum_{j=1}^{n} a_{kj} x_j - b_k \right| =$

min y

$$y \geq \sum_{j=1}^{n} a_{kj} x_j - b_k \quad \forall k = 1, 2, \cdots, H$$

$$y \geq -\left(\sum_{j=1}^{n} a_{kj} x_j - b_k\right) \quad \forall k = 1, 2, \cdots, H.$$

4). $\min \sum_{k=1}^{H} \left(\left| \sum_{j=1}^{n} a_{kj} x_j - b_k \right| \right) = \min \sum_{k=1}^{H} (y_k' + y_k'')$

$$y_k' - y_k'' = \sum_{j=1}^{n} a_{kj} x_j - b_k \quad \forall k = 1, 2, \cdots, H$$

$$y_i', y_i'' \geq 0 \quad \forall i.$$

5). Goal Programming

$\geq b_1 - y_1^-$

$\leq b_2 + y_2^+$ min $k_1 y_1^- + k_2 y_2^+ + k_3 y_3^- + k_4 y_3^+$.

$= b_3 - y_3^- + y_3^+$ Non-preemptive.

preemptive Goal programming

ignore lower, min higher, solve, fix.

6). Logic

$a \vee b$ becomes $x_a + x_b \geq 1$.

$a \wedge b$ becomes $x_a + x_b = 2$.

$\neg a$ becomes $x_a = 0$. $(1 - x_a) \Leftrightarrow \neg a$.

$a \Rightarrow b$ $x_a \leq x_b$

$a \Leftrightarrow b$ $x_a = x_b$

$c \Rightarrow a \vee b$. $x_c \leq x_a + x_b$

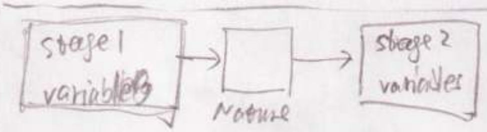
$c \Leftarrow a \vee b$. $x_c \geq x_a$, $x_c \geq x_b$

$c \Leftrightarrow a \vee b$. $x_c \geq x_a$, $x_c \geq x_b$, $x_c \leq x_a + x_b$

$c \Rightarrow a \wedge b$ $x_c \leq x_a$, $x_c \leq x_b$.

$c \Leftarrow a \wedge b$. $x_c \geq x_a + x_b - 1$

$c \Leftrightarrow a \wedge b$. $x_c \leq x_a$, $x_c \leq x_b$. $x_c \geq x_a + x_b - 1$.



SP: objective $\max \sum P(a^T x + b)$ → different a, b

RO: objective $\max \min (a^T x + b)$.

兩級策略.

7). $(a \wedge b \wedge \neg c) \vee (\neg a \wedge b \wedge \neg c) \Rightarrow (\neg a \wedge d \wedge e)$ is true.

x_1 x_2 x_3

$$\begin{cases} x_1 \le x_3 \\ x_2 \le x_3 \end{cases}$$

$$\begin{cases} x_1 \ge x_a + x_b + (1-x_c) - 2 \\ x_1 \le x_a \\ x_1 \le x_b \\ x_1 \le 1 - x_c \end{cases}$$

$$\begin{cases} x_2 \ge (1-x_a) + x_b + (1-x_c) - 2 \\ x_2 \le 1 - x_a \\ x_2 \le x_b \\ x_2 \le 1 - x_c \end{cases}$$

$$\begin{cases} x_3 \ge (1-x_a) + x_d + x_e - 2 \\ x_3 \le 1 - x_a \\ x_3 \le x_d \\ x_3 \le x_e \end{cases}$$

8). Switching constraints on/off

$\boxed{\text{Consider about the number of constraints}}$

$y=1 \Rightarrow a^T x \le b.$

$a^T x \le b + M(1-y)$

$y=1 \Rightarrow a^T x \ge b.$

$a^T x \ge b - M'(1-y).$

$y=1 \Rightarrow a^T x = b$

$$\begin{cases} a^T x \le b + M(1-y) \\ a^T x \ge b - M'(1-y). \end{cases}$$

$y=1 \Leftrightarrow a^T x \le b.$

$$\begin{cases} a^T x \le b + M(1-y) \\ a^T x \ge b + \varepsilon - M'y. \end{cases} \quad ①$$

$y=1 \Leftrightarrow a^T x \ge b$

$$\begin{cases} a^T x \ge b - M'(1-y) \\ a^T x \le b + M'y - \varepsilon. \end{cases} \quad ②$$

$y=1 \Leftrightarrow a^T x = b$

① + ②

calculate M : $\Rightarrow$ formulate to $M \ge (\cdots a \cdots b)$ max it

9). Branch & Bound.

$$IP_0 \rightarrow LP_0 \rightarrow \begin{cases} IP_1 \rightarrow LP_1 \quad \cdots \\ IP_2 \rightarrow LP_2 \end{cases}$$

(remember)

if it's min the lb is not the IP_0's lb, but IP_n's lb.

trick : ① use $y' - y'' = |\cdot\cdot|$, min $y' + y''$ or let $y' + y'' \leq A$ and $\Sigma y' + \Sigma y'' \ge B$

② $y \ge |a^T x| \Rightarrow \begin{cases} y \ge \Sigma a^T x \\ y \ge -\Sigma ax \end{cases} \xrightarrow{\text{switch on/off}} \begin{cases} y \ge \Sigma ax - M(1-z) \\ y \ge -\Sigma ax - M(1-z) \end{cases}$

maybe the same z.

10). $\max\{|x|, |y|\} \ge 1 \Rightarrow \begin{cases} x \ge 1 - My_1 \\ y \ge 1 - My_2 \\ -x \ge 1 - My_3 \\ -y \ge 1 - My_4 \\ \Sigma y_i \le 3. \end{cases}$