El método de simplex

Dpto. Ingeniería Industrial, Universidad de Chile

IN3701, Optimización

7 de marzo de 2011

Contenidos

Resumen

- Todo problema en forma estándar tiene puntos extremos.
- Si un problema en forma estándar tiene solución óptima, un punto extremo es solución óptima también.
- La idea central del método se basa en este hecho.
- Revisa condiciones de optimalidad, y si no es óptimo, se mueve a una nueva base factible de mejor costo.
- En esta sección consideramos el problema

$$(P) \quad \begin{array}{llcl} \text{mín} & cx & & \\ s.t. & Ax & = & b \\ & x & \geq & 0 \end{array}$$

- Denotaremos como $P := \{x \in \mathbb{R}^n_+ : Ax = b\}$, asumimos $A \in \mathbb{R}^{m \times n}$, $\mathrm{rank}(A) = m$, llamamos A_i la i-ésima columna, y a_i la i-ésima fila.

Condiciones de optimalidad

Definición 2.1

Dado $x \in P$, se dice que x es un óptimo local si existe $\varepsilon > 0$ tal que $\forall x' \in P \cap B(\varepsilon, x)$ se tiene que $cx' \geq cx$.

Teorema 2.1

Si $x \in P$ es óptimo local para (P), entonces es óptimo global.

- Esto implica que basta caracterizar óptimos locales.
- Podemos asumir que x es una solución básica factible.
- ¿Podemos dar una descripción algebraica?

Definición 2.2

Sea $x \in P$, un vector $d \in \mathbb{R}^n$ es una dirección admisible si existe $\theta > 0$ tal que $x + \theta d \in P$. El conjunto de todas las direcciones admisibles en un punto $x \in P$ se denota como $\mathcal{D}_P(x)$.

Caracterizando óptimos locales

- Suponemos $\overline{x} \in P$ solución básica factible, y sea B una base asociada.
  - Notemos que $\forall x' \in S := P \cap B(\varepsilon, \overline{x})$, tenemos que $d = \overline{x} - x' \in \mathcal{D}(\overline{x})$.
  - Es decir, basta caracterizar $\mathcal{D}(\overline{x})$ para caracterizar S.
  - Además si existe $x' \in S$ tal que $cx' < c\overline{x}$ tenemos que $cd < 0$.
  - Esto implica que óptimo local es equivalente a demostrar que $cd \geq 0 \forall d \in \mathcal{D}(\overline{x})$.
  - Pero $d \in \mathcal{D}(\overline{x})$ si y sólo si $Ad = 0$ y $\overline{x} + \theta d \geq 0$ para algún $\theta > 0$.
  - Usando el hecho que B es base, tenemos:

$$\begin{aligned} Ad &= 0 \quad \Leftrightarrow \\ (A_B, A_N)(d_B, d_N)^t &= 0 \quad \Leftrightarrow \\ d_B + \sum_{i \in N} \underbrace{A_B^{-1} A_i}_{\overline{A}_i} d_i &= 0 \end{aligned}$$

  - Fijando $d_i = 1$, $v^i = (v^i_B, v^i_N) = (-\overline{A}_i, e_i)$ satisface $Ad = 0$.
  - De donde todo $d \in \mathcal{D}(\overline{x})$ satisface $d = \sum(\lambda_i v^i : i \in N)$ con $\lambda_i \geq 0$.

Caracterizando óptimos locales

Definición 2.3

El vector $v^i = (v^i_B, v^i_N) = (-\overline{A}_i, e_i) : i \in N$, se llama i-ésima dirección básica en $\overline{x}$.

Teorema 2.2

Si $\overline{x}$ es una solución básica factible no degenerada, entonces $\mathcal{D}(\overline{x}) = \{\sum_{i\in N} \lambda_i v^i : \lambda_i \geq 0\}$. En general $\mathcal{D}(\overline{x}) \subseteq \{\sum_{i\in N} \lambda_i v^i : \lambda_i \geq 0\}$.

- Volviendo a la condición de optimalidad:
  - Notemos que para todo $d \in \mathcal{D}(\overline{x})$ se tiene que $cd = \sum(\lambda_i c v^i : i \in N)$ $cd = \sum(\lambda_i \underbrace{(c_i - c_B \overline{A}_i)}_{\overline{c_i}} : i \in N)$, con $\lambda_i \geq 0$.
  - En caso no degenerado, podemos concluir $cd \geq 0 \quad \forall d \in \mathcal{D}(\overline{x}) \iff \overline{c_i} \geq 0 \quad \forall i \in N$.

Caracterizando óptimos locales

Definición 2.4

Dada una solución básica factible x asociada a una base B, definimos el costo reducido de $x_i : i \in N$ como $\overline{c_i} = c_i - c_B\overline{A_i} = c_i - c_B A_B^{-1} A_i$.

Teorema 2.3

Dada una solución básica factible x asociada a una base B, y el vector de costos reducidos $\overline{c} \in \mathbb{R}^N$, entonces:

- Si $\overline{c} \geq 0$, entonces x es óptimo.
- Si x es no degenerado y óptimo, entonces $\overline{c} \geq 0$.

Observación 2.1

Dado x asociada a una base B, entonces:

$$(P) \begin{array}{ll} \min & cx \\ st. & Ax = b \\ & x \geq 0 \end{array} \quad \Leftrightarrow \quad (P') \begin{array}{ll} \min & c_B\overline{b} + \overline{c}x_n \\ st. & x_B + \overline{A_N}x_N = \overline{b} \\ & x \geq 0 \end{array}$$

Caracterizando óptimos locales

Definición 2.5

Una base B se dice óptima si:

1. $\overline{b} = A_B^{-1}b \geq 0.$
2. $\overline{c} = c_N - c_B\overline{A_N} = c_N - c_B A_B^{-1} A_N \geq 0.$

- Claramente la solución básica asociada a una base óptima es factible, y como satisface las condiciones del teorema anterior, es una solución óptima.
- No necesariamente una base asociada a una solución básica factible óptima es una base óptima.
- El algoritmo de simplex es capaz de resolver este problema.
- ¿Cuál es la interpretación geométrica de las direcciones básica?
- ¿Que interpretación tienen los $\overline{c_i} : i \in N$ en general?
- Dada una solución básica factible, ¿cómo demostramos optimalidad y/o encontramos una mejor solución?

El Método de Simplex

- Consideremos B base factible y $\overline{x}$ la s.b.f asociada.
- Si $\overline{c} \geq 0$ B es una base óptima.
- Si no, sea $i \in N : \overline{c_i} < 0$, y consideremos $(P_i) : \min\{c'(\overline{x} + \lambda v^i) : A(\overline{x} + \lambda v^i) = b, \overline{x} + \lambda v^i \geq 0\}$.
- Por construcción (P_i) es equivalente a $\min\{\lambda\overline{c}_i : \overline{x_B} - \lambda\overline{A}_i \geq 0\}$.
- Si $(\overline{A}_i)_j \leq 0$, $\forall j = 1...m$, (P_i) (y (P)) es no acotado.
- Definiendo $\lambda^* = \min\{\overline{x_{B_j}}/(\overline{A}_i)_j : (\overline{A}_i)_j > 0\}$, (P_i) puede escribirse como $\min\{\lambda\overline{c}_i : 0 \leq \lambda \leq \lambda^*\} = \lambda^*\overline{c}_i$.
- Sea $B_i^* = \{j \in B : \overline{x_j} - \lambda^*(\overline{A}_i)_j = 0, (\overline{A}_i)_j > 0\}$ y sea $j \in B_i^*$.
- Si $B_{ij} = B \cup \{i\} \setminus \{j\}$ es una base, podemos re-comenzar el proceso.
- Si todos los vértices son no-degenerados, el proceso mejora continuamente la función objetivo.
- Si (P) admite solución óptima, el algoritmo encontrará la s.b.f. óptima.

El Método de Simplex

Teorema 2.4

Para todo $i \in N$, y $j \in B_i^*$ el conjunto B_{ij} define una base factible de (P). y $\overline{x} + \lambda^* v^i$ es la solución básica asociada.

- Dem: use el hecho que existe μ tal que $A_B \mu = A_i$.

Algoritmo de Simplex v1.0:

Input: una solución básica factible B.

1. Computar $y = c_B A_B^{-1}$.
2. Si $\forall i \in N$, $\overline{c_i} = c_i - yA_i \geq 0$, B es base óptima, return.
3. Escoger $i \in N : \overline{c_i} < 0$ y computar $\overline{A_i} = A_B^{-1} A_i$.
4. Si $B_i^* = \phi$ el problema es no acotado, return.
5. Escoger $j \in B_i^*$.
6. $B \leftarrow B_{ij}$, volver a 1.

El Método de Simplex

- La variable $i \in N$ del paso tres se llama variable entrante.
- La variable $j \in B_i^*$ del paso cinco se llama variable saliente.
- Reglas para escoger i, j en pasos 3, 5, se llaman reglas de pivoteo.
- Si (P) es no-degenerado y factible, el algoritmo termina con una solución al problema.
- Si el algoritmo termina, termina con la respuesta correcta.
  - Incluso si (P) posee vértices degenerados.
- ¿Qué posibles reglas de pivoteo hay?
  - mínimo $\overline{c_i}$, Steepest descend, first negative, round-robin.
  - ¿Hay reglas que aseguren que el algoritmo siempre termine?
- ¿Cuánto tiempo demora cada iteración?
  - Resolver dos sistemas $yA_B = c_B$ y $\overline{A_i}A_B = A_i$.
  - realizar entre 1 y n multiplicaciones de vectores de tamaño m.
- ¿Cuántas iteraciones son necesarias?
  - Entre una, hasta número de vértices de (P).

Manejando Degenerancia en Simplex

Definición 2.6

Dado $u, v \in \mathbb{R}^n$, decimos que u es lexicográficamente mayor que v ($u \overset{L}{\geq} v$) si la primera componente no-cero de $u - v$ es positiva.

- Note que $\overset{L}{\geq}$ es un orden total en $\mathbb{R}^n$.
- Regla lexicográfica:
  - Escoger $i \in N$ con mayor impacto en función objetivo.
  - Si todas tienen impacto cero, escoger $i \in N$ con $\overline{c_i} < 0$.
  - Consideremos $u = \overline{A_i}$, escogemos variable saliente j tal que $\overline{b_j}/u_j$ es mínimo para los $u_j > 0$.
  - j esta únicamente determinada, por que? (rango A).

Teorema 2.5

El algoritmo de Simplex bajo la regla lexicográfica satisface:

- El vector $(-c_B\overline{b}, c - c_B A_B^{-1} A) \in \mathbb{R}^{n+1}$ incrementa estrictamente (en el orden lexicográfico) en cada iteración.
- El algoritmo siempre termina.

Buscando una base inicial

- Hasta ahora hemos demostrado que dada una base inicial factible de (P), el algoritmo de simplex termina con una solución al problema.
- ¿Cómo encontramos una base inicial?
- Supongamos que $b \geq 0$ y consideremos

$$(P_o): \quad \left\{ \text{mín} \sum_{i=1}^{m} y_i : Ax + y = b, (x, y) \geq 0 \right\}.$$

- Claramente (P_o) es factible, y tiene solución óptima.
- Además $z_{P_o} = 0 \iff P$ es factible.
- Variables y se llaman artificiales, y siempre están asociadas a una base factible de (P_o).
- Resolver (P_o) es lo que se llama hacer Fase I de Simplex.

Algoritmo de Simplex

Algoritmo de Simplex v2.0:

Dado $(P): \{\min cx : Ax = b, x \geq 0\}$ con $b \geq 0$.

- Definir $(P_o): \{\min \mathbb{1}y : Ax + y = b, (x, y) \geq 0\}$.
- Utilizar Simplex v1.0 con $B = \{n+1, \ldots, n+m\}$.
- Si $z_{P_o} > 0$ (P) es infactible, return.
- Definir $(P'): \{\min cx + \infty\mathbb{1}y : Ax + y = b, (x, y) \geq 0\}$.
- Utilizar Simplex v1.0 con B la base óptima de (P_o).
- Si no acotado, return.
- Reportar z_P, base óptima B y x^*, return.

- Nóte que utilizar $\infty\mathbb{1}y$ en la función objetivo, es equivalente a asumir que $\overline{c_i} > 0$ para toda variable $y_i \in N$.
- Ésta es la implementación práctica mas usual.

Eficiencia computacional de Simplex:

- Una pregunta central es ¿Cuántas iteraciones de simplex son necesarias?
  - Considere el ejemplo $P = \{x \in \mathbb{R}^n : 0 \leq x_i \leq 1\}$.
  - P tiene 2^n vertices.
  - Es posible deformar P de forma tal de armar un camino que recorra todos los vertices incrementando estrictamente la función objetivo.
  - Para la mayoría de las reglas de pivoteo conocidas hay deformaciones adecuadas que fuerzan al algoritmo a visitar todos los vertices.
- ¿Qué pasa en la práctica? ¿En promedio?
  - Difícil caracterizar todos los problemas prácticos.
  - Más aun definir una probabilidad de distribución.
  - Estudios empíricos muestran que el número de iteraciones es $\mathcal{O}(m\log(n))$.
  - ¿Existen algoritmos mas rápidos siempre?
    - Algoritmo del Elipsoide.
    - Algoritmos de Punto Interior.