IE 495 – Lecture 10

Properties of the Recourse Function

Prof. Jeff Linderoth

February 12, 2003

Today's Topic

M

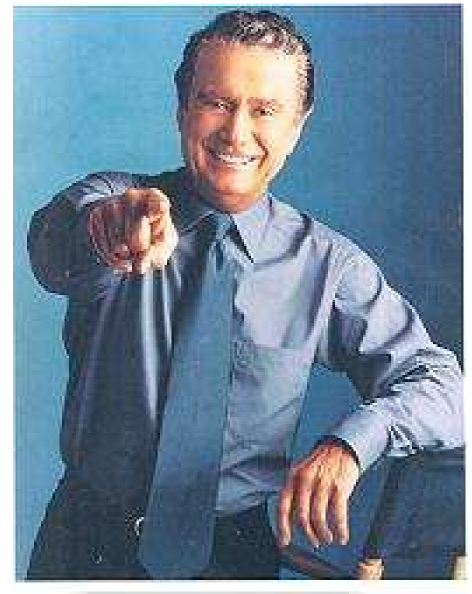
E

T

H O D

Outline

- Small amount of review
- KKT/Optimality conditions for two-stage stochastic LP w/recourse
- The LShaped algorithm
  - Example



Is That Your Final Answer

- What is the subgradient inequality?
- What are the KKT conditions?
- $Q(x, \omega) = \min_{y \in \Re^p_+} \{q^T y : Wy = h(\omega) - T(\omega)x\}$.
  - ◇ Name a vector $s \in \partial Q(x, \omega)$.
- $\mathcal{Q}(x) = \mathbb{E}_\omega Q(x, \omega) = \sum_{s \in S} p_s Q(x, \omega)$.
  - ◇ Name a vector $s \in \mathcal{Q}(x)$.
- What is the Deterministic Equivalent of a stochastic program?

Our Favorite Problem

$$\min_{x \in \Re^n_+ : Ax = b} \left\{ c^T x + \mathbb{E}_\omega \left[\min_{y \in \Re^p_+} \{ q^T y : W y = h(\omega) - T(\omega) x \} \right] \right\}$$

$$\min_{x \in \Re^n_+ : Ax = b} \left\{ c^T x + \mathbb{E}_\omega Q(x, \omega) \right\}$$

$$\min_{x \in \Re^n_+} \{ c^T x + \mathcal{Q}(x) : Ax = b \}$$

Highlights from Chapter 9

- $Q(x, \omega) \equiv v(h(\omega) - T(\omega)x)$ is convex.
- $\mathcal{Q}(x) \equiv \mathbb{E}_\omega Q(x, \omega)$ is convex
- $\mathcal{Q}(x)$ is Lipschitz-continuous.
- If $\min_{y \in \Re^p_+} \{q^T y : Wy = h(\omega) - T(\omega)x\}$ has unique dual solution λ^*, then $\nabla Q(x, \omega) = -\lambda^{*T} T$
- If $\mathcal{Q}(x) = \sum_{s \in S} p_s Q(x, \omega_s)$, then

$$\eta = -\sum_{s \in S} p_s \lambda_s^{*T} T(\omega_s) \in \partial \mathcal{Q}(x)$$

Continuous Discussion

- Computing $\mathcal{Q}(x) = \int_\Omega Q(x,t)dF(t)$ in general requires numerical integration for a given value of x
- Computing $\nabla \mathcal{Q}(x)$ also would require numerical integration.

⋆ This is only possible when ω is a vector of very small dimensionality.

- Typically people (and we will too) discretize the continuous distribution.
  - ◇ We'll talk about this later (soon, actually)...

KKT Conditions

Here, again for your convenience are the KKT conditions (in their non-differentiable extension).

- **Thm:** For a convex function $f : \Re^n \mapsto \Re$, and convex functions $g_i : Re^n \mapsto \Re, i = 1, 2, \ldots m$, if we have some nice "regularity conditions" (which we have in this case), $\hat{x}$ is an optimal solution to $\min_{x \in \Re^n_+} \{f(x) : g_i(x) = 0 \ \forall i = 1, 2, \ldots m\}$ if and only if the following conditions hold:
  - ◇ $g_i(x) = 0 \quad \forall i = 1, 2, \ldots m$
  - ◇ $\exists \lambda_1, \lambda_2, \ldots \lambda_m \in \Re, \mu_1, \mu_2, \ldots \mu_n \in \Re_+$ such that
    - $0 \in \partial f(\hat{x}) + \sum_{i=1}^m \lambda_i \partial g_i(\hat{x}) - \sum_{j=1}^n \mu_j$.
    - $\mu_j \geq 0 \ \forall j = 1, 2, \ldots n$
    - $\mu_j \hat{x}_j = 0 \ \forall j = 1, 2, \ldots n$

Apply to Our Problem

$$\min_{x \in \Re^n_+} \{c^T x + \mathcal{Q}(x) : Ax = b\}$$

Thm: $\hat{x} \in K_1$ is optimal if and only if

- $\exists \lambda \in \Re^m, \mu \in \Re^n_+$ such that
  - $0 \in c + \partial \mathcal{Q}(\hat{x}) + A^T \lambda - \mu$
  - $\mu^T \hat{x} = 0$

Or

$$-c - A^T \lambda + \mu \in \partial \mathcal{Q}(\hat{x})$$

2-Stage SLP. Deterministic Equivalent

- If the two stage SLP comes from a probability space with finite support...

$$
\begin{array}{ccccccccccc}
c^T x & + & p_1 q^T y_1 & + & p_2 q^T y_2 & + & \cdots & + & p_s q^T y_s & & \\
\text{s.t.} & & & & & & & & & & \\
Ax & & & & & & & & & = & b \\
T_1 x & + & W y_1 & & & & & & & = & h_1 \\
T_2 x & & & + & W y_2 & & & & & = & h_2 \\
\vdots & & & + & & & \ddots & & & & \\
T_S x & & & & & & & + & W y_s & = & h_s \\
x \in X & & y_1 \in Y & & y_2 \in Y & & & & y_s \in Y & &
\end{array}
$$

About the DE

- $y_s \equiv y(\omega_s)$ is the recourse action to take if scenario ω_s occurs.
- Pro: It's a linear program.
- Con: It's a BIG linear program.
  - ◇ $n + pS$ variables
  - ◇ $m_1 + mS$ constraints.
- Pro: The matrix of the linear program has a very special (staircase) structure.
  - ? Has anyone heard of Bender's Decomposition?

The L-Shaped Method

- Bender's decomposition applied to the DE.
- However, we're going to think of it as a subgradient-based optimization method
  - ◇ We spent *lots* of time showing $\mathcal{Q}(x) = \sum_{s \in S} p_s Q(x, \omega_s)$ was convex (and nondifferentiable)
  - ◇ $\mathcal{Q}(x)$ is piecewise linear, with lots of bumps.

A Picture

LShaped Method

- We know that a subgradient of $\mathcal{Q}(x)$ $\hat{x}$ looks like...

$$u = -\sum_{s \in S} p_s \lambda_s^* T_s \in \partial \mathcal{Q}(\hat{x}),$$

- where λ^* is an optimal dual solution to the recourse problem in scenario s:

$$\lambda_s^* = \arg\max_{\lambda} \{\lambda^T (h_s - T_s \hat{x}) : \lambda^T W \leq q\}.$$

LShaped Method

- So that by the subgradient inequality...

$$\mathcal{Q}(x) \geq \mathcal{Q}(\hat{x}) + u^T(x - \hat{x})$$

- In other words $\mathcal{Q}(\hat{x}) + u^T(x - \hat{x})$ is a supportins hyperplane of $\mathcal{Q}$ at $\hat{x}$.
- This insight is used to build up an (increasingly better) approximation of $\mathcal{Q}(x)$.

LShaped Method

- Let the variable θ be our approximation to the function $\mathcal{Q}(x)$...
- For any (feasible) value of x $\theta \approx \mathcal{Q}(x)$.

? Now how do we go about building such an approximation θ?

  ⋆ Use the subgradient inequality!

LShaped Method

- Imagine that we had L subgradients of $Q(x)$
- $u_1 \in \partial \mathcal{Q}(x_1), u_2 \in \partial \mathcal{Q}(x_2), \ldots u_l \in \partial \mathcal{Q}(x_l)$
- Then...

minimize

$$c^T x + \theta$$

subject to

$$\begin{aligned} Ax &= b \\ \theta &\geq \geq \mathcal{Q}(x_l) + u_l^T (x - x_l) \qquad \forall l = 1, 2, \ldots L \end{aligned}$$

Example: Our Favorite Random Linear Program

minimize

$$x_1 + x_2$$

subject to

$$\begin{aligned} \omega_1 x_1 + x_2 &\geq 7 \\ \omega_2 x_1 + x_2 &\geq 4 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

- $\omega = (\omega_1, \omega_2) \in \Omega = \{(1, 1/3), (5/2, 2/3), (4, 1)\}$

Example Worked out...

Next time

- More LShaped...
  - ◇ Formal algorithm specification
  - ◇ Correctness/Convergence
  - ◇ Implementing in AMPL
  - ◇ Infeasibility cuts
  - ◇ Multicut methods
- Homework #2. And this time I *MEAN* it!
  - ◇ Homework will be on the website Friday if you want to get started over the weekend.