Linear Programming
Sensitivity Analysis

Sensitivity Analysis

- Rationale
- Shadow Prices
  - Definition
  - Use
  - Sign
  - Range of Validity
- Opportunity Costs
  - Definition
  - Use

Rationale for Sensitivity Analysis

- **Math problem is an approximation**
  - optimum is an approximation
  - we need to check
- **Constraints often artificial**
  - Designer should question them
  - *Should we have different specifications?*
- **Situations always probabilistic**
  - Prices change
  - Need to assess risk

Shadow Price Definition

- **Recall from Constrained Optimization:**
  - **Shadow price = δ (objective function)/ δ (constraint) at the optimum**
  - **Complementary Slackness:**
    Either (Slack variable) or (shadow price) = 0

Shadow Price Illustration

Max: $Y = X_1 + 4X_2$

s.t. $X_1 + X_2 \leq 5 = b_1$

$X_1 \geq 3 = b_2$

$X_2 \leq 3 = b_3$

$X_1 , X_2 \geq 0$

Notes:

a) $X_1^* = 3$; $X_2^* = 2$; $Y^* = 11$

b) when $\Delta b_1 = \pm 1$

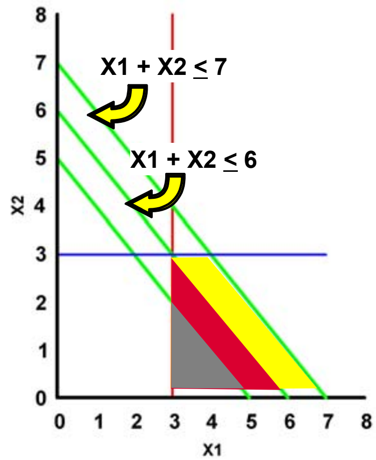
$\Delta X_2^* = \pm 1$; $\Delta Y^* = \pm 4$; $SP_1 = 4$

c) $SP_3^* = 0$; $slack_3 = 1$

d) when $b_1 > 6$

$slack_3 = 0$; $SP_3 \neq 0$;

$SP_1 = 1 \leq 4$



Proactive Use of Shadow Prices

- Identify constraints with high S.P
- See if they can be changed for better solutions
- Example: New York water supply
  - Original Design for Third City Tunnel ($1 billion plus)
  - pressure < 40 psi at curb (some point in Brooklyn)
  - No allowance for local tanks, pumps
  - Shadow price in millions of dollars!

Reactive Use of Shadow Prices

- Respond to new opportunities
- Example: client changes specifications

- Respond to proposals for new constraints
- Example: trace chemicals

Sign of Shadow Prices

- "Obvious Rule" (+SP with +Δb) not correct
- Correct Reasoning:
  - What makes the optimum better?
    - Expansion of feasible region => "Relaxation of constraints"
  - What changes will increase the feasible region?
    - Increase upper bound $\Sigma_j \, a_{ij}X_j < b_i$
    - Decrease lower bound $\Sigma_k \, a_{kj}X_j > b_k$
  - i.e., "Raise the roof, lower the floor."

Shadow Price Illustration

Max: $Y = X_1 + 4X_2$

s.t.
$X_1 + X_2 \le 5 = b_1$
$X_1 \ge 3 = b_2$
$X_2 \le 3 = b_3$
$X_1, X_2 \ge 0$

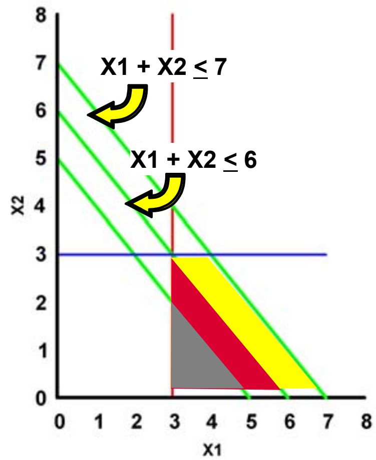
Notes:

a) $X_1^* = 3$; $X_2^* = 2$; $Y^* = 11$

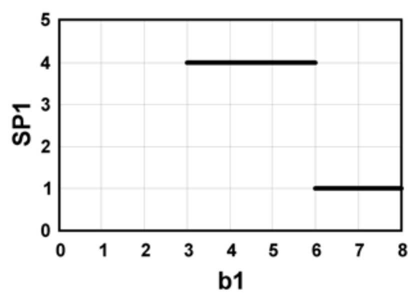
b) when $\Delta b_1 = \pm 1$
$\Delta X_2^* = \pm 1$; $\Delta Y^* = \pm 4$; $SP_1 = 4$

c) $SP_3^* = 0$; $slack_3 = 1$

d) when $b_1 > 6$
$slack_3 = 0$; $SP_3 \neq 0$;
$SP_1 = 1 \le 4$



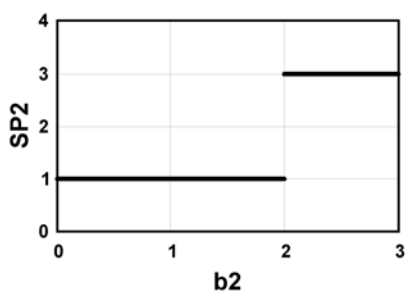
Shadow Prices As Constraints Change



increase an
upper bound
("raise the roof")

decrease a lower bound
("lower the floor")

b_2: 3 --> 2
new $\underline{X}^*$ = [2,3]
new Y^* = 14
$DY^* = 3$



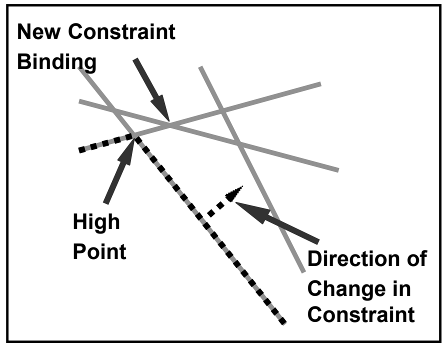
Range of Shadow Prices

- In Linear Programming, Shadow prices are constant
- Until a constraint changes enough so that a new constraint is binding
- Results given as:

SP_K = constant

for $r_L < b_K < r_U$

- Outside the range:
  - –Shadow prices <u>decrease</u> as constraint is relaxed
  - –Shadow prices <u>increase</u> as constraint is tightened



Shadow Price Ranges for Example

Max: $Y = X_1 + 4X_2$

s.t. $X_1 + X_2 \leq 5 = b_1$

$X_1 \geq 3 = b_2$

$X_2 \leq 3 = b_3$

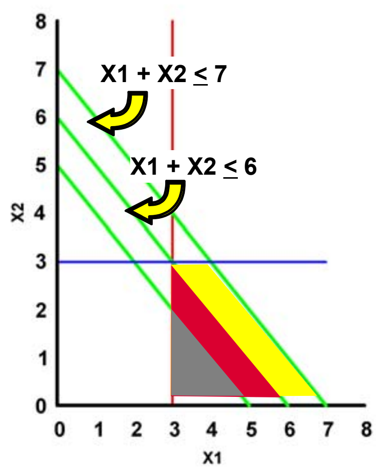
$X_1, X_2 \geq 0$

Shadow Prices

$SP_1 = 4 \qquad 3 \leq b_1 \leq 6$

$SP_2 = 4 \qquad 2 \leq b_2 \leq 5$

$SP_3 = 0 \qquad 2 \leq b_3$



Opportunity Cost - Definition

- Objective Function = $\Sigma c_i X_i$
- Opportunity costs associated with c_i -- the coefficients of design/decision variables
- At optimum, some decision variables = 0
  - These are non-optimal decision variables
- Opportunity cost is:
  - Degradation of optimum per unit of non-optimal variable introduced into design
  - A "cost" in that it is a worsening of optimum. Units may be almost anything; equal to whatever units are being optimized.

Meaning of Opportunity Costs

- Opportunity cost defines design trigger "price"
  - The value of the coefficient of the decision variable for which that variable should be in the design
- Suppose: Obj.Function = ... + $c_K X_K$ + ...

  and X_K not optimal with an opportunity cost = OC_K
- Then, as c_K changes for the better, (greater for maximization, lesser for minimization)
  - OC_K lower
  - $OC_K = 0$ at $c_K' = c_K - OC_K$
- c_K' is trigger price; defines the limit of best design

Illustration of Opportunity Cost

- What happens when forced to use a non-optimal decision variable?
- Example: Min Cost = $2X_1 + 10X_2 + 20 X_3$

$$\text{s.t.} \quad X_1 + X_2 + X_3 \geq 3$$
$$X_2 \geq 1$$
$$X_1, X_2, X_3 \geq 0$$

- $\underline{X}^* = (2, 1, 0)$; cost* = 14
- If forced to use X_3, new X* = (1,1,1); new cost* = 32

  Thus: (opportunity cost)$_3$ = $\Delta Z^*/1 = 18$

Use of Opportunity Cost

- At what price would it be desirable to use X_3 ?
- If X_3 is used with no change in its unit cost (= c_3), the optimal cost would increase by 18
- If the cost of X_3 were to fall by an amount equal to the opportunity cost ($c_3' = c_3 - OC_3 = 20 - 18 = 2$). It would then compete with X_1
- So the answer is: When its unit cost falls by its opportunity cost: 20 - 18 = 2

How do you find SP and OC?

- LP optimization programs all calculate shadow prices and opportunity costs routinely and "print them out" for you

- Sometimes, programs report this information in special ways. Thus:
  - Shadow Prices < = > "dual decision variables"
  - Opportunity Costs < = > "dual slack variables"
  - More on this later

A Possible Semantic Confusion

- Note that the Phrases "shadow price" and "opportunity cost" have somewhat different meanings in LP and Economics literature

- The "opportunity cost" of an action in economics can be interpreted as the "shadow price" of that action on the budget…

Summary on LP Sensitivity Analysis

- LP Optimization Programs automatically provide important information useful for improving/changing design
- Shadow prices -- to help redefine constraints
- Opportunity costs -- to identify critical prices

- Need to understand these quantities carefully