MAT 361 – Operations Research
Brooks Emerick

CHAPTER 6 – Sensitivity Analysis and Duality

Section 6.1: A Graphical Introduction to Sensitivity Analysis

- Sensitivity analysis is the study of how changes in parameters of an LP problem affect the LP's optimal solution.
- Parameters are essential coefficients in the constraints or in the objective function. We'll study two scenarios:

   1.) Objective Function Coefficients: for what range of values can an obj. fun. coefficient range while maintaining optimality in the same set of basic variables? This is called the optimality range.

   2.) Constraint Right-Hand Side: for what range of values can we vary the RHS of a constraint while maintaining optimality in the same set of basic variables? This is called the feasibility range.

   ↳ Leads to the shadow price – the rate of change of the obj. fun. value per unit change of a resource.

- Graphically, changing the coefficients c_1, c_2 in the objective function $Z = c_1 x_1 + c_2 x_2$ is equivalent to changing the slope of the isoprofit/isocost line. Consider the following:

- Let z = fixed, then

$$z = c_1 x_1 + c_2 x_2$$

$$\Rightarrow \quad x_2 = -\frac{c_1}{c_2} x_1 + \frac{z}{c_2}$$

$$\Rightarrow \quad y = -\frac{c_1}{c_2} x + \frac{z}{c_2}$$

c_2 affects y-intercept.

Both c_1 and c_2 affect slope.

- For what threshold values of c_1, c_2 with the set of basic variables at optimality change?

$\boxed{\text{WS}}$ #1 JOBCO example to illustrate Sensitivity.

Section 6.2: Some Important Formulas

- Before we get into Sensitivity Analysis, we need to (yet again) cast the LP into a matrix form.
- Assume we are solving a Max problem that has been prepared using any method previously discussed (ie. Big M method) where there are n decision variables, $x_1, x_2, \ldots, x_n$, (note: x_i may include slack, excess, and artificial variables) and m constraints.
- Define the following:
  - $\vec{C} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$ = column vector of obj. fun. coefficients $(n \times 1)$

- $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ = column vector of decision variables $(n \times 1)$

- $A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$ = coefficients of equality constraints, $(m \times n)$

- $\vec{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$ = vector of right-hand side values

- With these definitions we define the max LP as

$$\boxed{\begin{aligned} &\text{Maximize } z = \vec{c}^T \vec{x} \\ &\text{Subject to: } A\vec{x} = \vec{b} \\ &\qquad\qquad\quad \vec{x} \geq \vec{0} \end{aligned}}$$

- For any set of basic variables (i.e. a set of m variables out of the possible n decision variables), we can decompose this system into "basic" and "non-basic" parts. Define the following:
  - Let BV be the set of m indices that correspond to the basic variables.

    $$BV = \{BV_1, BV_2, \ldots, BV_m\}$$

    so that $\{x_{BV_1}, x_{BV_2}, \ldots, x_{BV_m}\}$ are the basic vars.
  - Likewise, NBV is the set of $n-m$ indices for the nonbasic variables

    $$NBV = \{NBV_1, NBV_2, \ldots, NBV_{n-m}\}$$

- Create two vectors from $\vec{x}$ that consist of the Basic and Non-Basic variables:

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} \qquad \vec{x}_{BV} = \begin{bmatrix} x_{BV_1} \\ x_{BV_2} \\ \vdots \\ x_{BV_m} \end{bmatrix} \ (m \times 1) \text{ vector}$$

order doesn't really matter

$$\vec{x}_{NBV} = \begin{bmatrix} x_{NBV_1} \\ x_{NBV_2} \\ \vdots \\ x_{NBV_{n-m}} \end{bmatrix} \ ((n-m) \times 1) \text{ vector}$$

- Similarly, split the obj. fun. coeff vector into Basic and Non-Basic parts:

$$\vec{c} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix} \qquad \vec{c}_{BV} = \begin{bmatrix} c_{BV_1} \\ c_{BV_2} \\ \vdots \\ c_{BV_m} \end{bmatrix} \ (m \times 1) \text{ vector}$$

order must be the same as $\vec{x}_{BV}, \vec{x}_{NBV}$

$$\vec{c}_{NBV} = \begin{bmatrix} c_{NBV_1} \\ c_{NBV_2} \\ \vdots \\ c_{NBV_{n-m}} \end{bmatrix} \ ((n-m) \times 1) \text{ vector}$$

- Form two matrices, B and N, from the matrix A where the columns of B are created from the columns of A corresponding to the indices in the set BV. Likewise for N with the set NBV:

$$A = \begin{array}{cccccc} BV_1 & NBV_1 & BV_2 & \cdots & NBV_{n-m} & BV_m \\ \left[\begin{matrix} a_{11} \\ a_{21} \\ \vdots \\ a_{m1} \end{matrix}\right. & \begin{matrix} a_{12} \\ a_{22} \\ \vdots \\ a_{m2} \end{matrix} & \begin{matrix} a_{13} \\ a_{23} \\ \vdots \\ a_{m3} \end{matrix} & \begin{matrix} \cdots \\ \cdots \\ \ddots \\ \cdots \end{matrix} & \begin{matrix} a_{1,n-1} \\ a_{2,n-1} \\ \vdots \\ a_{m,n-1} \end{matrix} & \left.\begin{matrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{mn} \end{matrix}\right] \end{array}$$

$$B = \begin{bmatrix} a_{11} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m3} & \cdots & a_{mn} \end{bmatrix} \qquad N = \begin{bmatrix} a_{12} & \cdots & a_{1,n-1} \\ a_{22} & \cdots & a_{2,n-1} \\ \vdots & \ddots & \vdots \\ a_{m2} & \cdots & a_{m,n-1} \end{bmatrix}$$

$$(m \times m) \qquad\qquad (m \times (n-m))$$

- Using these definitions, we can write the original LP in the following terms:

$$\left.\begin{array}{ll} \text{Maximize} & z = \vec{c}^T\vec{x} \\ \text{Subject to:} & A\vec{x} = \vec{b} \\ & \vec{x} \geq \vec{0} \end{array}\right\} \rightarrow \left\{\begin{array}{ll} \text{Maximize} & \vec{z} = \vec{C}_{BV}^T \vec{X}_{BV} + \vec{C}_{NBV}^T \vec{X}_{NBV} \\ \text{Subject to} & B\vec{X}_{BV} + N\vec{X}_{NBV} = \vec{b} \\ & \vec{X}_{BV}, \vec{X}_{NBV} \geq \vec{0} \end{array}\right.$$

- Convince yourself that these indeed are equivalent.

$\sqrt{\boxed{\text{WS}}}$ #1 work on deconstructing the LP.

- Now, we note that we can "solve" for the basic variables by multiplying the left-hand side of the constraint system by B^{-1}:

$$B\vec{X}_{BV} + N\vec{X}_{NBV} = \vec{b}$$

$$\Rightarrow \quad B^{-1}(B\vec{X}_{BV} + N\vec{X}_{NBV}) = B^{-1}\vec{b}$$

$$\Rightarrow \quad \boxed{\vec{X}_{BV} + B^{-1}N\vec{X}_{NBV} = B^{-1}\vec{b}}$$

Note: since $\vec{X}_{NBV} = \vec{0}$, we have the optimal solution as $\vec{X}_{BV} = B^{-1}\vec{b}$.

This matrix system represents the tableau corresponding to this set of basic variables

- Since $\vec{X}_{BV}$ can be found using the Simplex Algorithm, this equation above represents the optimal tableau and the matrix B (and B^{-1}) is unique to this set of basic variables.

$\boxed{\text{WS}}$ #2 write LP solution in matrix/vector form.

- we can also Row 0 in the optimal tableau in matrix/vector form. First, consider the following

$$\vec{X}_{BV} = -B^{-1}N\vec{X}_{NBV} + B^{-1}\vec{b}$$

Z_{opt}:

$$\underbrace{\vec{C}_{BV}^{\,T}\vec{X}_{BV}} = -\vec{C}_{BV}^{\,T}B^{-1}N\vec{X}_{NBV} + \vec{C}_{BV}^{\,T}B^{-1}\vec{b}$$

Also, we have

$$Z = \vec{C}_{BV}^{\,T}\vec{X}_{BV} + \vec{C}_{NBV}^{\,T}\vec{X}_{NBV}$$

$$\Rightarrow Z = -\vec{C}_{BV}^{\,T}B^{-1}N\vec{X}_{NBV} + \vec{C}_{BV}^{\,T}B^{-1}\vec{b} + \vec{C}_{NBV}^{\,T}\vec{X}_{NBV}$$

$$\Rightarrow \underbrace{Z + \left(\vec{C}_{BV}^{\,T}B^{-1}N - \vec{C}_{NBV}^{\,T}\right)\vec{X}_{NBV}}_{\text{LHS of Row 0 at optimal tbl}} = \underbrace{\vec{C}_{BV}^{\,T}B^{-1}\vec{b}}_{\text{RHS of Row 0 i.e. optimal value of obj. function}}$$

$\boxed{\text{WS}}$ #3 work on the Row 0 of optimal tableau.

- Focus on the term in parentheses above:

$$\underbrace{\left(\vec{C}_{BV}^{\,T}B^{-1}N - \vec{C}_{NBV}^{\,T}\right)}_{\text{row vector of size } 1\times m}\vec{X}_{NBV}$$

This row vector is made up of the coefficients of the non-basic variables (the basic variable coefficients are zero). Since N is made up of columns of A, we define

$$\bar{c}_j = \vec{c}_{BV}^{\,T} B^{-1} \vec{a}_j - c_j$$

Coefficient of non-basic variable j in Row 0 under current set BV.

j^{th} column of matrix A

Original Row 0 coefficient of variable j

WS New LP. Use $\bar{c}_j$ to determine optimality range

Note: If we're trying to determine the $\bar{c}_i$ for a non-basic slack variable that is denoted by s_i, then its original A column, denoted $\vec{a}_i$, will have the form

$$\vec{a}_i = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \leftarrow i^{th} \text{ element.}$$

and $c_i = 0$ (it contributes nothing to the objective function).

Hence,

$$\bar{c}_i = \vec{c}_{BV}^{\,T} B^{-1} \vec{a} - c_i$$

$$\Rightarrow \bar{c}_i = \vec{c}_{BV}^{\,T} B^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} - 0$$

$$\Rightarrow \boxed{\bar{c}_i = i^{th} \text{ element of } \vec{c}_{BV}^{\,T} B^{-1}}$$

Section 6.3: Sensitivity Analysis

- we now look at several cases in which changing specific coefficients in the LP will have an impact on the feasibility and/or optimality.

1.) Changing the objective value coefficient of a basic variable:

$\boxed{\text{WS}}$ #1a, #1b. Gripetto Problem Sensitivity.

- General Technique:
    1.) obtain an optimal basis via simplex method. Identify B^{-1} and N.
    2.) Select an objective function coefficient to study, e.g.

$$Z = 3x_1 + 2x_2$$

OLD: $\vec{C}_{BV}^T = [3 \quad 2 \quad 0]$

define $C_1 = 3 + \Delta C_1$, where ΔC_1 denotes a deviation of that coefficient.

NEW: $\vec{C}_{BV}^T = [3+\Delta C_1 \quad 2 \quad 0]$

    3.) Compute $C_{BV}^T B^{-1}$ and identify the $\bar{C}_j$ values (Row 0 coefficients) under this deviation.
    4.) Determine the optimality range by considering the values of ΔC_2 that maintain $\bar{C}_j \geq 0$ (for max LP) or $\bar{C}_j \leq 0$ (for min LP).

- Outcomes / Effects:
    1.) Entire Row 0 may change in optimal tableau
    2.) For C_1 in the optimality range, we know Row 0 coefficients for NBVs are non-negative (for max) or non-positive (for min)
    3.) For C_1 in the optimality range, basic variable values will not change, but the optimal z-value will change e.g.

OLD: $Z = 3x_1 + 2x_2$

NEW: $Z = 3 + 3\Delta C_1 x_1 + 2x_2$.

New optimal value $Z_{opt} = [3+\Delta C_1 \quad 2 \quad 0] B^{-1}\vec{b}$
with same optimal solution $\vec{X}_{BV} = B^{-1}\vec{b}$.

2.) Changing the RHS coefficient of a constraint.

$\boxed{\text{WS}}$ #1c, #1d Gerapetto Problem Sensitivity

- General Technique:
  1.) obtain an optimal basis via simplex method. Identify B^{-1} and $\vec{b}$.
  2.) Select a RHS constraint coefficient to study, e.g.

OLD: $\vec{b} = \begin{bmatrix} 100 \\ 80 \\ 40 \end{bmatrix}$ new: $\vec{b} = \begin{bmatrix} 100+\Delta b_1 \\ 80 \\ 40 \end{bmatrix}$

define $b_1 = 100 + \Delta b_1$, where Δb_1 is some deviation of that value.

  3.) Compute $B^{-1}\vec{b}$ to find the new basis values in terms of this deviation.
  4.) Determine the feasibility range by considering the values of Δb_1 that maintain $B^{-1}\vec{b} \geq \vec{0}$ i.e. every "new" basis value must be non-negative.

- Outcomes / Effects
  1.) Row 0 in optimal tableau does not change.
  2.) For b_1 in feasibility range, we know the original set of optimal values $\vec{X}_{BV}$ is still feasible and optimal. The values of $B^{-1}\vec{b}$ are all non-negative.
  3.) For b_1 in feasibility range, the basis of BVs do not change but their values do as well as the optimal value of the objective function.

BVs OLD: $\vec{X}_{BV} = B^{-1}\begin{bmatrix} 100 \\ 80 \\ 40 \end{bmatrix}$ z_{opt} OLD: $z_{opt} = \vec{C}_{BV}^{\,T} B^{-1} \begin{bmatrix} 100 \\ 80 \\ 40 \end{bmatrix}$

NEW: $\vec{X}_{BV} = B^{-1}\begin{bmatrix} 100+\Delta b_1 \\ 80 \\ 40 \end{bmatrix}$ NEW: $z_{opt} = \vec{C}_{BV}^{\,T} B^{-1} \begin{bmatrix} 100+\Delta b_1 \\ 80 \\ 40 \end{bmatrix}$

3.) Adding a new activity

$\boxed{WS}$ #1a, #1b Grapetto Problem Sensitivity

- General Technique:

  1.) obtain an optimal basis to the original problem via simplex method. Identify B^{-1} and $\vec{c}_{BV}$.

  2.) Creating a new activity adds an obj. function coefficient, c_j, and a new column to the matrix A, say $\vec{a}_j$. Price out this new variable, x_j, by computing the Row 0 coefficient under the current optimal basis:

$$\bar{c}_j = \vec{c}_{BV}^{\,T} B^{-1} \vec{a}_j - c_j$$

  3.) If $\bar{c}_j$ is negative (for max LP) or positive (for min LP), then it is beneficial to consider the new activity.

- Outcomes/Effects

  1.) Changes will affect the Row 0 coefficient of the new variable x_j in the optimal tableau.

  2.) Values of current optimal basis and objective function will not change since we're not adding new variable to basis (yet)

  3.) The current basis will still be optimal if $\bar{c}_j \geq 0$ (for max) and $\bar{c}_j \leq 0$ (for min).

  4.) If $\bar{c}_j < 0$ (for max) or $\bar{c}_j > 0$ (for min), then this new variable will enter basis and so should be added to the problem.

Section 6.4: Changing More than One Parameter and The 100% Rule

- In some instances we may want to consider what happens to the optimality of our LP under several parameter changes.

- We can determine if the optimality stays the same under a variety of changes using the <u>100% Rule</u>, which considers the optimality and feasibility ranges of several coefficients at once.

- Changing several objective function coefficients:
  - Suppose we wish to implement changes in each of the coefficients c_j for $j = 1, 2, \ldots, n$. Then we define the following:

$$c_j = \text{original coeff value}$$
$$\Delta c_j = \text{deviation of the } j^{th} \text{ coeff value}$$
$$I_j = \text{allowable increase, i.e. upper bound on } \Delta c_j \text{ found through sensitivity analysis.}$$
$$D_j = \text{allowable decrease, i.e. lower bound on } \Delta c_j \text{ found through sensitivity analysis.}$$

  - Using the values of I_j and D_j, we compute the ratio or proportion of allowable deviation:

$$\text{If } \Delta c_j \geq 0, \text{ define } r_j = \frac{\Delta c_j}{I_j}$$

$$\text{If } \Delta c_j \leq 0, \text{ define } r_j = -\frac{\Delta c_j}{D_j}$$

Note: by definition of optimality range, we have that the current basis is still optimal if

$$D_j \le \Delta c_j \le I_j$$

$$\Rightarrow \quad -1 \le r_j \le 1$$

for any single value j.

- The 100% Rule:

If $\sum_{j=1}^{n} r_j \le 1$, then the current basis remains optimal.

- The same rule applies for changing any number of RHS values of the constraints, say b_j, for $j = 1, 2, \dots, n$.

$\boxed{\text{Pros}}$ #1 a,b Use Excel output to implement 100% Rule.