Linear-Programming duality

Advanced Algorithms (CSE 794)
Instructor: Tamal K. Dey
Scribe: Wei Liu *

April 30, 2009

1 The reason why study LP duality

In the previous lecture, we know that Simplex Algorithm could terminate, with returning a feasible solution or non-feasible one. However, we do not know whether that feasible solution is optimal. Therefore, the concept of linear-programming duality is used to prove that it is the optimal solution.

As is said in the textbook, duality is a very important property, and it is powerful in its ability to provide a proof that a solution is indeed optimal. My understanding of its insight is that if one problem (maximum) could be changed to another (minimum) which has related coefficients, then if these two problems satisfy two conditions: one is that result of the first one is no more then the second one, the other one is that we could find one solution satisfying both problems, which means both of these two problems could get the same result, then we could say this solution is the optimal one for both of them. This is obvious; the maximum problem could not get a greater value then this optimal one since if there is then it contradicts the first condition.

2 Comparing Standard LP form with Dual form

Standard LP form: Maximize: $\sum_{j=1}^{n} c_j x_j$

s.t. $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ for $i = 1, 2, \ldots, m$ and $x_j \geq 0$ for $j = 1, 2, \ldots, n$

Dual form: Minimize: $\sum_{i=1}^{m} b_i y_i$

s.t. $\sum_{i=1}^{m} a_{ij} y_i \geq c_j$ for $j = 1, 2, \ldots, n$ and $y_i \geq 0$ for $i = 1, 2, \ldots, m$

In Standard LP: n is the variables' number and m is the constraints' number. In Dual form: n is the constraints' number and m is the variables' number. We may find here that the dual form actually is a new linear programming problem with the same coefficients of the standard LP form (And of course the coefficients exchanged with each other).

*Department of Computer and Information Sciences, The Ohio State University, Columbus, Ohio, USA.

3 The proof

Lemma 1. *Let $\overline{x}$ and $\overline{y}$ be any feasible solutions to LP and its dual. Then:*

$$\sum_{j=1}^{n} c_j\overline{x}_j \leq \sum_{i=1}^{m} b_i\overline{y}_i$$

Proof:

$$\begin{aligned}
\sum_{j=1}^{n} c_j\overline{x}_j &\leq \sum_{j=1}^{n}(\sum_{i=1}^{m} a_{ij}y_i)\overline{x}_j \qquad \text{(by } c_j \leq \sum_{i=1}^{m} a_{ij}y_i \text{ in Dual form expression)} &(1)\\
&\leq \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij}\overline{x}_j)\overline{y}_i &(2)\\
&\leq \sum_{i=1}^{m} b_i\overline{y}_i \qquad \text{(by } \sum_{j=1}^{n} a_{ij}x_j \leq b_i \text{ in Standard LP expression)} &(3)
\end{aligned}$$

Corollary 2. *Let $\overline{x}$ be a feasible solution to a primal LP (A, b, c), and let $\overline{y}$ be a feasible solution to the corresponding dual program, if $\sum_{j=1}^{n} c_j\overline{x}_j = \sum_{i=1}^{m} b_i\overline{y}_i$, then $\overline{x}$ and $\overline{y}$ are optimal solutions.*

The proof of this corollary is obvious, from *Lemma 1* we know that the LP's value is no more then Dual's value, so if there is some solutions for LP and Dual satisfying this equality, we can find other solution for LP, which makes its value greater then this one, it must be greater then some Dual's value, which contradicts the *Lemma 1*, so is Dual's solution. Therefore $\overline{x}$ and $\overline{y}$ are optimal solutions.

Suppose the last slack form returned from Simplex Algorithm is: $z = v^{'} \sum_{j\in N} c_j^{'}x_j$ and $x_i = b_i^{'} - \sum_{j\in N} a_{ij}^{'}x_j$ for $i \in B$, then an optimal dual solution is:

$$\overline{y}_i = \begin{cases} -c_{n+i}^{'} & : \quad \text{if } (n+i) \in N \\ 0 & : \quad \text{otherwise} \end{cases}$$

We can get $\overline{y}_i$'s value directly by Dual form expression.

Theorem 3. *Suppose Simplex returns $\overline{x} = (\overline{x}_2, \overline{x}_2, \ldots, \overline{x}_n)$ on the primal LP. Let $\overline{y}$ be defined as mentioned. Then $\overline{x}$ is an optimal solution to the primal and $\overline{y}$ is an optimal solution to the dual and:*

$$\sum_{j=1}^{n} c_j\overline{x}_j = \sum_{i=1}^{m} b_i\overline{y}_i$$

Proof: By corollary and $\overline{y}$'s definition.

$$z = v^{'} + \sum_{j\in N} c_j^{'}x_j \qquad c_j^{'} \leq 0 \text{ for } j \in N; v^{'}, c_j^{'} \text{ correspond to optimal solution} \qquad (4)$$

$$\text{Then define: } c_j^{'} = 0 \qquad \text{for} j \in B \qquad \text{For basic variable x, its coefficient is 0} \qquad (5)$$

$$z = v^{'} + \sum_{j=1}^{n+m} c_j^{'}x_j \qquad (6)$$

$$\sum_{j=1}^{n} c_j\overline{x}_j = v^{'} + \sum_{j=1}^{n+m} c_j^{'}\overline{x}_j \qquad \text{left side is initial objective function, right is the final one} \qquad (7)$$

$$\text{For any set of values} \qquad x = (x_1, x_2, \ldots, x_n) \qquad (8)$$

$$
\begin{aligned}
\sum_{j=1}^{n} c_j x_j &= v' + \sum_{j=1}^{n+m} c_j' x_j \qquad \text{True since objective function is equivalent with others.} && (9)\\
&= v' + \sum_{j=1}^{n} c_j' x_j + \sum_{j=n+1}^{n+m} c_j' x_j && (10)\\
&= v' + \sum_{j=1}^{n} c_j' x_j + \sum_{i=1}^{m} c_{i+n}' x_{i+n} && (11)\\
&= v' + \sum_{j=1}^{n} c_j' x_j + \sum_{i=1}^{m} (-\overline{y}_i) x_{i+n} \qquad \overline{y}_i\text{'s definition} && (12)\\
&= v' + \sum_{j=1}^{n} c_j' x_j + \sum_{i=1}^{m} (-\overline{y}_i)(b_i - \sum_{j=1}^{n} a_{ij} x_j) \text{ Definition of basic variable in LP} && (13)\\
&= v' + \sum_{j=1}^{n} c_j' x_j - \sum_{i=1}^{m} \overline{y}_i b_i + \sum_{i=1}^{m}\sum_{j=1}^{n} (a_{ij} x_i)\overline{y}_i && (14)\\
&= (v' - \sum_{i=1}^{m} \overline{y}_i b_i) + \sum_{j=1}^{n} (c_j' + \sum_{i=1}^{m} a_{ij}\overline{y}_i) x_j && (15)\\
So \quad v' &= \sum_{i=1}^{m} \overline{y}_i b_i \qquad \text{and } c_j = c_j' + \sum_{i=1}^{m} a_{ij}\overline{y}_i \le \sum_{i=1}^{m} a_{ij}\overline{y}_i && (16)
\end{aligned}
$$

Here, we use a obvious truth that if there is a form: $\sum_{j=1}^{n} c_j x_j = v + \sum_{j=1}^{n} b_j x_j$, then $v = 0$ and $c_j = b_j$ for every j.

4 Feasible solution evaluation

There is one more problem left, in the beginning of Simplex algorithm, we use a sub procedure Initialize-Simplex (A, b, c) to determines whether the input, linear program, has any feasible solutions, and if there is it will return a slack form(N, A, B, b, c, v).

But, how does this Initialize-Simplex work, if a linear program is feasible, then there are two situations, the first one is that the initial basic solution is feasible, then the procedure Initialize-Simple will do few things to convert the problem to slack form. However, there exists another situation, the initial basic solution may not be feasible, but the linear program could have a feasible solution, or not. Like this example:

$$
\begin{aligned}
\text{maximize: } 2x_1 - x_2 & && (17)\\
2x_1 - x_2 &\le 2 && (18)\\
x_1 - 5x_2 &\le -4 && (19)\\
x_1, x_2 &\ge 0 && (20)
\end{aligned}
$$

It does not have a initial basic solution, but it has a feasible solution. So Initial-Simplex need to do more stuff.

Lemma 4. *Let L be an LP in standard form, let L_{aux} be an auxiliary linear program:*

$$\text{maximize:} \quad -x_0 \tag{21}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_{ij}x_j - x_0 \leq b_i \quad \text{for } i = 1, \ldots, m \tag{22}$$

$$x_j \geq 0 \quad \text{for } j = 1, \ldots, n \tag{23}$$

Then L is feasible iffL_{aux} has optimal value of 0. (24)

Proof: If L has a feasible solution $\overline{x}$, then $\overline{x}$ and $\overline{x}_0$=0 together will be a feasible solution to L_{aux} with objective value 0. And because the constraint of L_{aux} has $-x_0 \geq 0$ and its objective function is $-x_0$, so we know its max objective value is zero. Conversely, if the optimal objective value of L_{aux} is 0, then $\overline{x}_0 = 0$is easy to find, and the values of the remaining variables trivially satisfy the constraints of L.

Initialize-Simplex(A, b, c):

Let l be the index with minimal b_l

If $b_l \geq 0$, then return $(1, 2, \ldots, n, n+1, \ldots, n+m, A, b, c, 0)$, this means that we could find a initial basic solution.

If not, we will run PIVOT on L_{aux}, for L_{aux} by adding $-x_0$. Let (N, B, A, b, c, v) be resulting slack form of $L_{aux}(N, B, A, b, c, v) \leftarrow$ PIVOT$(N, B, A, b, c, v, l, 0)$

Iterate the while loop of Simplex until an optimal solution of L_{aux} is obtained, if the basic solution sets $\overline{x}_0 = 0$, then return the find slack form with x_0 removed and the original objective function restored.

Else, return Infeasible

5 Conclusion

What I learnt: First, the main point of this class is to prove the Simplex algorithm could get a optimal solution. Also with a tricky auxiliary linear program L_{aux}, we could determine whether a linear program has a feasible solution.

Second, I think the hard part is to design the dual form, as in this problem, we need to carefully make a dual form so that we can get proper $\overline{y}_i$.

Last, need notice the function Initialize-Simplex(A, b, c), when the linear program does not have initial basic solution, but it does have a feasible solution, then in this function we could not just simply convert (A, b, c) to $(N, B, A, b, c,)$, instead, we should use the result of L_{aux} to modify the input slack form of that linear program.