The Simplex Algorithm

Andrea Gagliano, Melissa Jones, Gary Morales, Brian Purviance

June 6, 2013

Abstract

This document utilizes Sage to implement George Bernard Dantzig's simplex algorithm of maximization/minimization for a given linear equation subject to specific linear constraints. Primal, and duality methods are applied as well as Bland's rule for pivoting to avoid algorithmic cycling. This is implementation in meant as a tutorial.

1 Introduction

Dantzig's simplex algorithm is applied to systems of this type:
The equation to manipulate (maximize/minimize) is your objective function.
$c_1x_1 + c_2x_2 = y$
subject to constraint equations;
$a_1x_1 + a_2x_2 \leq b_1$
$a_1x_1 + a_2x_2 \leq b_2$

2 Tableau

2.1 Initial Tableau (Primal & Dual Methods)

Primal method is described as:
Objective equation $\mathbf{C^T x}$
Subject to constraints: $\mathbf{Ax} \leq \overrightarrow{\mathbf{b}}$

Dual method is described as:

Table 1: Primal Method

0	$\mathbf{A}$	$\mathbf{I}$	$\overrightarrow{\mathbf{b}}$
-1	$\mathbf{C^T}$	0	0

Objective equation $\mathbf{b^T y}$
Subject to constraints: $\mathbf{A^T y} \leq \overrightarrow{\mathbf{C}}$

When implementing the simplex algorithm, the use of a tableau (canonical) is required to process through to a solution. Its initial general set up is as follows:

The far left column is not necessarily needed; it simply is used to maintain sign for your objective value. The final row is the most important as it tracks how your objective function matures to a feasible solution. Example:

	x_1	x_2	x_3	x_4	
0	1	1	1	0	10
0	1	2	0	1	20
-1	5	3	0	0	0

Here our $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$. And $\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, the equivalently sized identity matrix. $\overrightarrow{\mathbf{b}} = \begin{bmatrix} 10 & 20 \end{bmatrix}$. The identity matrix then becomes your basis variables as you manipulate through the algorithm. Finally $\mathbf{C^T} = \begin{bmatrix} 5 & 3 \end{bmatrix}$. The solution to this example tableau is:

	x_1	x_2	x_3	x_4	
0	1	1	1	0	10
0	0	1	-1	1	10
-1	0	-2	-5	0	-50

Here one feasible solution to this problem is:
$\mathbf{x_1} = 10$
$\mathbf{x_2} = 0$
And your objective value (maximum point) is 50.

Table 2: Dual Method

0	$\mathbf{A^T}$	$\mathbf{I}$	$\overrightarrow{\mathbf{C}}$
-1	$\mathbf{b^T}$	0	0

Dual method implementation is similar to above. It simply starts as described in Table 2

Note: Every pivot move is a feasible solution to the optimization problem however the max/min has not been reached until every value in the objective row is either a negative value or a zero.

2.2 Pivots & Pivoting

When initiating the simplex algorithm, a pivot position must first be established.

	x_1	x_2	x_3	x_4		ratios
0	1	1	1	0	10	10/1
0	1	2	0	1	20	20/1
-1	5	3	0	0	0	

Starting from this initial tableau a new column titled 'ratios,' has been created which is done by dividing the values in the column associated with greatest value in the $\mathbf{C^T}$ row by those in the $\overrightarrow{\mathbf{b}}$. In this case it is the column associated with the $\mathbf{x_1}$ variable, thus defining your pivot column. The pivot position is selected from the smallest ration created. Then the task simply becomes row reducing your $\mathbf{A}$ matrix to the identity (similar to inverting a matrix).
Note: ratio column is only used to select your pivot position.
This type of pivoting/reducing is completed when your objective function row becomes a combination of negatives and zeros.

2.3 Bland's Rule

In order to ensure a feasible solution is reached, this method of anti-cycling has been implemented. Cycling happens when your tableau's objective function never transforms itself to negative and zero values. It entails simply choosing the left most column $\mathbf{x_1}$ to create ratios versus the $\overrightarrow{\mathbf{b}}$. Then choosing your pivot position based on the smallest ratio created.