Optimisation and Operations Research

Lecture 5: The Simplex Algorithm

Matthew Roughan
<matthew.roughan@adelaide.edu.au>
http://www.maths.adelaide.edu.au/matthew.roughan/notes/OORII/

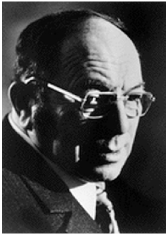
School of Mathematical Sciences,
University of Adelaide

August 13, 2019

Section 1

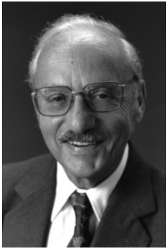
Simplex overview

A little history



Leonid Vitaliyevich Kantorovich;
(19 January 1912 – 7 April 1986) was a Soviet mathematician and economist, known for his theory and development of techniques for the optimal allocation of resources. He was the winner of the Nobel Prize in Economics in 1975 and the only winner of this prize from the USSR.

George Bernard Dantzig:
(November 8, 1914 – May 13, 2005) was an American mathematical scientist who made important contributions to operations research, computer science, economics, and statistics. He is particularly well known for his development of the simplex algorithm and his work with linear programming, some years after it was pioneered by the Soviet mathematician and economist Leonid Kantorovich.



Problem Recap

We will start with a problem in *standard equality form*, find $\mathbf{x}$ that solves

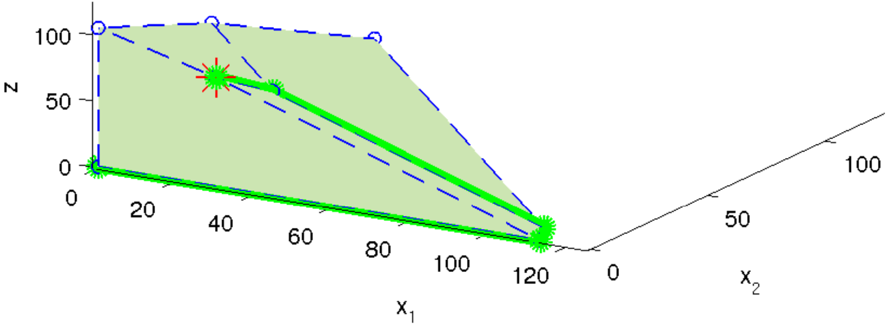
$$\begin{array}{rrcl} \max & z & = & \mathbf{c}^T\mathbf{x} + z_0 \\ \text{such that} & A\mathbf{x} & = & \mathbf{b} \\ \text{and} & \mathbf{x} & \geq & 0 \end{array}$$

or in abbreviated form

$$\underset{\mathbf{x}}{\operatorname{argmax}} \left\{ z = \mathbf{c}^T\mathbf{x} + z_0 \,\middle|\, A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0 \right\}$$

Intuition

- Remember that often this has been converted from inequality form: so the vector $\mathbf{x}$ could include slack variables from inequalities.
- Intuitively, we want to search the vertices (efficiently) of the original feasible region for the optimal vertex.
- Vertices (of the original inequality problem) are feasible basic solutions to the equality
  - We will organise our search so as to force the constraints towards being true, and then staying true
  - Search always improves the objective



Our First Problem

$$
\begin{array}{rcrcrcrcl}
\max & z = & 13x_1 & + & 12x_2 & + & 17x_3 & & \\
\text{subject to} & & 2x_1 & + & x_2 & + & 2x_3 & \leq & 225 \\
 & & x_1 & + & x_2 & + & x_3 & \leq & 117 \\
 & & 3x_1 & + & 3x_2 & + & 4x_3 & \leq & 420 \\
 & & & & & & x_i & \geq & 0, \quad \text{for } i = 1, 2, 3
\end{array}
$$

Simplex Overview

1. Form the Simplex Tableau
   - it's convenient to put all the information in one large matrix
2. Phase I
   - look for a starting point
   - either we get
     - ★ a feasible starting point
     - ★ we show the problem is infeasible
3. Phase II
   - find the optimal point
   - or show that the problem is unbounded

Teaching Order

We teach this in the order

1. Tableau
2. Phase II
3. Phase I

because

- sometimes Simplex Phase I isn't needed
- phase I can be written in terms of Phase II

Section 2

The Simplex Tableau

The Simplex Tableau

The Simplex Tableau M for

$$\operatorname*{argmax}_{\mathbf{x}} \left\{ z = \mathbf{c}^T\mathbf{x} + z_0 \,\middle|\, A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0 \right\}$$

is constructed by

		z col.	b col.	
$M =$	A	$\mathbf{0}$	$\mathbf{b}$	
	$-\mathbf{c}^T$	1	z_0	z row

- Assuming there are m constraints, and n variables, the Tableau M has size $(m+1) \times (n+2)$.
- Note the z column never changes, so we don't need to actually have this, but its useful to understand the algorithm.

The Simplex Tableau Example

Example

The LP from our first problem in standard equality form

$$\begin{array}{llcccccccl}
\max\ z = & 13x_1 + 12x_2 + 17x_3 + 0 \\
\text{subject to} \\
& 2x_1 + x_2 + 2x_3 + 1x_4 & = 225 \\
& x_1 + x_2 + x_3 + 1x_5 & = 117 \\
& 3x_1 + 3x_2 + 4x_3 + 1x_6 & = 420 \\
& x_i \geq 0 \quad \forall i
\end{array}$$

can be written in Tableau form (with $z_0 = 0$) as

x_1	x_2	x_3	x_4	x_5	x_6	z	b
2	1	2	1	0	0	0	225
1	1	1	0	1	0	0	117
3	3	4	0	0	1	0	420
−13	−12	−17	0	0	0	1	0

Why use the Simplex Tableau?

The Simplex Tableau isn't really a necessary part of the algorithm

- *e.g.*, sometimes people put the various bits in different parts of the Tableaux – it doesn't matter

So why do it?

- The Simplex Tableau puts all the data in one place
  - useful back in the days of crappy memory management
- Does it in a way that means we can use a single operation
  - `pivot` does the bulk of the work
  - makes the code simpler
- It's useful for teaching
  - we can understand what the algorithm is doing in a very simple way

The Simplex Tableau and solutions

Imagine the Simplex Tableau

$$M = \begin{array}{|c|c|c|} \hline A & \mathbf{0} & \mathbf{b} \\ \hline -\mathbf{c}^T & 1 & z_0 \\ \hline \end{array}$$

can be (re)written in the form

$$M' = \begin{array}{|c|c|c|} \hline Q & I & \mathbf{b}' \\ \hline \end{array}$$

Then we can read off the solution immediately:

- set the free variables $x_i = 0$ for $i = 1, \ldots, n$
- then just read off the basic variables $x_{n+i} = b'_i$ for $i = 1, \ldots, m$

We want to generalise and exploit this idea

The Simplex Tableau and Solutions

Example

x_1	x_2	x_3	x_4	x_5	x_6	z	b
2	1	2	1	0	0	0	225
1	1	1	0	1	0	0	117
3	3	4	0	0	1	0	420
−13	−12	−17	0	0	0	1	0

Solution:

$$\mathbf{x} = \big(0, 0, 0, 225, 117, 420\big) \text{ and } z = 0$$

We want to generalise and exploit this idea:

- we are looking for *unit* columns, *i.e.*, columns with m entries zero and the remaining element is 1.
- we need a complete set of these unit columns

Basic columns

- If x_j is a basic variable, then the jth column $M(:, j)$ will be a unit column, which we call a *basic column*
- A solution should have m basic variables, so we need m basic columns, plus 1 (the z column).
- The basic columns need to have their '1's in different rows, or we don't really have independent basic variables.

Let's construct a vector of the indexes of the basic variables:

$$\ell_B = [\ell_1, \ell_2, \ldots, \ell_m]$$

then we want to get M in the form such that when we take the columns corresponding to these variables, we get an identity matrix:

$$\left[M_{\ell_1} \; M_{\ell_2} \; \ldots \; M_{\ell_m} \; M_z \right] = I_{m+1}$$

where I_{m+1} is the $(m+1) \times (m+1)$ identity matrix.

Canonical form

Definition (Canonical form)

The tableau M is said to be in *canonical form* if there is a complete set of $(m+1)$ unit columns:

$$\left[M_{\ell_1} \; M_{\ell_2} \; \ldots \; M_{\ell_m} \; M_z \right] = I_{m+1}$$

and $\ell_B = [\ell_1, \ell_2, \ldots, \ell_m]$ is the canonical form of the list of basic variables.

Example

Note that the basic variables don't have to be in order

$M =$

x_1	x_2	x_3	x_4	x_5	z	b
0	1	0	6	1	0	2
2	−3	1	1	0	0	5
0	−1	0	0	0	1	4

is in canonical form, with $\ell_B = [5, 3]$

Canonical form

Example

$M =$

x_1	x_2	x_3	z	b
1	2	0	0	2
0	3	0	0	3
0	0	1	0	4
0	4	0	1	4

is not in canonical form, because there is no unit column with 1 in the second row.

Feasible Canonical form

Definition (Feasible Canonical form)

A tableau M in canonical form is said to be in *feasible canonical form* iff its solution is feasible.

- We can detect a feasible canonical form by noting that the solution where we read of the x_i values as above will have x_i values that are either 0 or b_j. These automatically satisfy $A\mathbf{x} = \mathbf{b}$, but must also satisfy $\mathbf{x} \geq \underline{0}$, so

  a canonical form is feasible iff the $\mathbf{b}$ *column is non-negative!*[1]
- A feasible canonical form corresponds to a vertex of the original inequality feasible region.

[1] Ignoring the z_0 element

Getting to feasible canonical form

- Simplex Phase II starts with a matrix in feasible canonical form
  - how do we get there?
- If we start with the problem $\max_{\mathbf{x}} \left\{ z = \mathbf{c}^T \mathbf{x} + z_0 \,\middle|\, A'\mathbf{x}' \leq \mathbf{b}, \mathbf{x}' \geq 0 \right\}$, with $\mathbf{b} \geq 0$, then conversion to standard equality form will introduce slack variables that create the required identity matrix.

Example

x_1	x_2	x_3	s_4	s_5	s_6	z	b
2	1	2	1	0	0	0	225
1	1	1	0	1	0	0	117
3	3	4	0	0	1	0	420
−13	−12	−17	0	0	0	1	0

  - this canonical form corresponds to a starting point $\mathbf{x}' = \underline{0}$ (with positive slack variables), *i.e.*, the origin
- Otherwise we use Simplex Phase I to get into feasible canonical form before we perform Phase II.

Takeaways

- The Tableau M encapsulates all of the information we need to keep track of
  - we need to know how to construct it
- The form we aim for or want to keep is *feasible canonical form*
  - complete set of unit columns
  - $\mathbf{b}$ column is non-negative
- In this form we can read off a basic feasible solution

Further reading I