Lecture 9: Linear Programming

- A common optimization problem involves finding the maximum of a linear function of N variables

$$Z = \sum_{i=1}^{N} a_i \, x_i \qquad \text{(the "objective function")}$$

where the x_i are all non-negative

Linear Programming

….subject to a series of M constraints

$$\sum_{i=1}^{N} c_{ij}\, x_i \begin{Bmatrix} \leqslant \\ = \\ \geqslant \end{Bmatrix} b_j \qquad j = 1, 2, 3, \ldots, M$$

where the b_j are all non-negative

Linear Programming

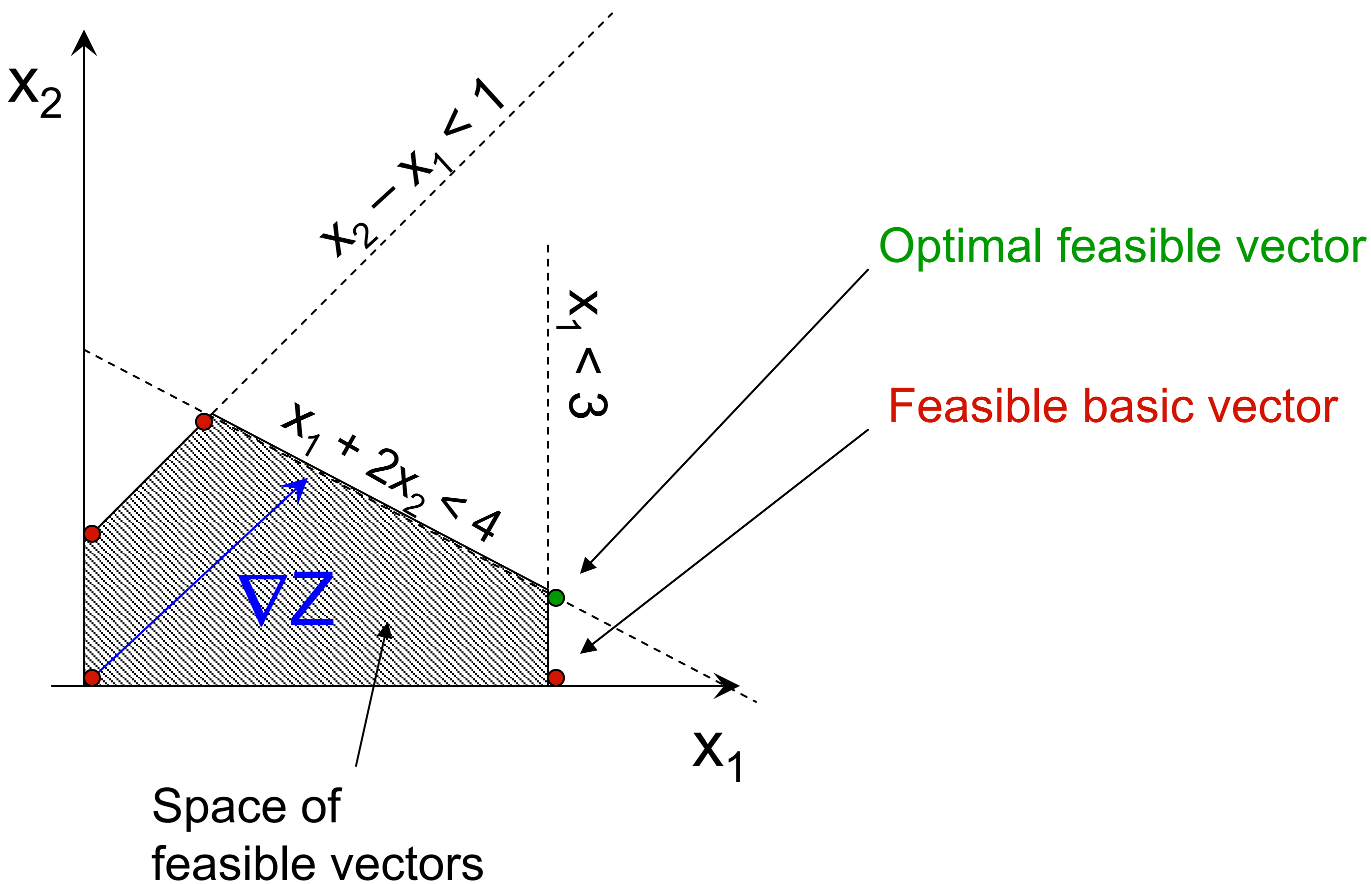
- The constraints are each represented by N–1 dimensional hyperplanes within the N-dim space
  - these form a convex hyperpolygon in N dimensions bounding a region of volume V^N
  - if V^N is zero (all volume eliminated), there is no solution
  - if V^N is > 0, then the gradient $\nabla Z = \sum_{i=1}^{N} a_i \mathbf{e_i}$ leads us to a boundary plane
    - Unless ∇Z is perpendicular to the boundary plane, we can follow that plane to an edge and that edge to a vertex ➔ a single optimal vector

Feasible vectors

- The space of vectors satisfying the constraints is called the space of “feasible vectors”, and the vertex that maximizes Z is the “optimal feasible vector”
- Vertices of the hyperpolygon are places that satisfy N constraints as equalities: these are called basic feasible vectors. (Here, the “constraints” include the non-negativity requirement on the x_i).

Feasible vectors

- 2-D example: $Z = x_1 + x_2$



The fundamental theorem of linear programming

"If an optimal feasible vector exists, then there is a feasible basic vector that is optimal"

i.e. the optimal vector is at one of the vertices

Our goal is to find which vertex (i.e. which N of the constraints does it satisfy)

Feasible basic vectors

The total number of locations at which N constraints are satisfied simultaneously can be very large:

It can be as large as the number of ways of choose N items out of N+M, which comes to $(N+M)! / (N!\ M!)$

In the 2-D example, $N=2$, $M=3$, and so

$(N+M)! / (N!\ M!) = 5!/(3!2!) = 10$

of which 5 lie in the feasible space and one is missing (since $x_1=0$ and $x_1=3$ cannot be satisfied simultaneously)

Restricted normal form

- The constraints can always be written such that
  - All constraints are written as equalities
  - Number of constraints $M \leqslant N$
  - Each constraint equation has a variable that appears in that equation alone

  (How to do this to be discussed later)
- This permits us to obtain a solution via the "simplex method"

The simplex method
(illustrated by example)

- Example (N=4, M=2)

Maximize $z = -4x_1 - 25x_2 + 4x_3 + x_4$

subject to
$$x_1 + 6x_2 - x_3 = 2$$
$$-3x_2 + 4x_3 + x_4 = 8$$

Move variables that appear in only one constraint equation to the left hand side and call them left hand variables

Solution via the simplex method

This yields

$$x_1 = 2 - 6x_2 + x_3$$
$$x_4 = 8 + 3x_2 - 4x_3$$

which we substitute into the objective function to obtain the latter in terms of right hand variables

$$z = 2x_2 - 4x_3$$

There are M left hand variables and N – M right hand variables

Solution via the simplex method

Represent these equations by a "tableau":

$$z = \quad 2x_2 - 4x_3$$

$$x_1 = 2 - 6x_2 + \quad x_3$$

$$x_4 = 8 + 3x_2 - 4x_3$$

is written

	const	x_2	x_3
z	0	+ 2	– 4
x_1	+ 2	– 6	+ 1
x_4	+ 8	+ 3	– 2

Solution via the simplex method

Setting the RH variables equal to zero, we can immediately obtain a basic feasible vector (but not, in general, the optimal one) x = (+2, 0, 0, +8)

Increasing x_3 from zero will clearly decrease z, so we conclude that the optimal feasible vector must have $x_3 = 0$

	const	x_2	x_3
z	0	+ 2	– 4
x_1	+ 2	– 6	+ 1
x_4	+ 8	+ 3	– 2

Solution via the simplex method

We now “pivot” about the negative entry that limits how large x_2 can become, and switch the “handedness” of x_1 and x_2.

$x_1 = 2 – 6x_2 + x_3$ ➔ $x_2 = 1/3 – x_1/6 + x_3/6$

$x_4 = 8 + 3x_2 – 4x_3$ ➔ $x_4 = 8 + 3(1/3 – x_1/6 + x_3/6) – 4x_3$

$= 9 – x_1/2 – 7x_3/2$

$z = 2x_2 – 4x_3$ ➔ $z = 2(1/3 – x_1/6 + x_3/6) – 4x_3$

$= 2/3 – x_1/3 – 11x_3/3$

Solution via the simplex method

Our new LH variables are x_1 and x_3 and our new RH variables are x_4 and x_1

And our new tableau is

	const	x_1	x_3
z	+2/3	– 1/3	– 11/3
x_2	+1/3	– 1/6	+ 1/6
x_4	+ 9	– 1/2	– 7/2

Solution via the simplex method

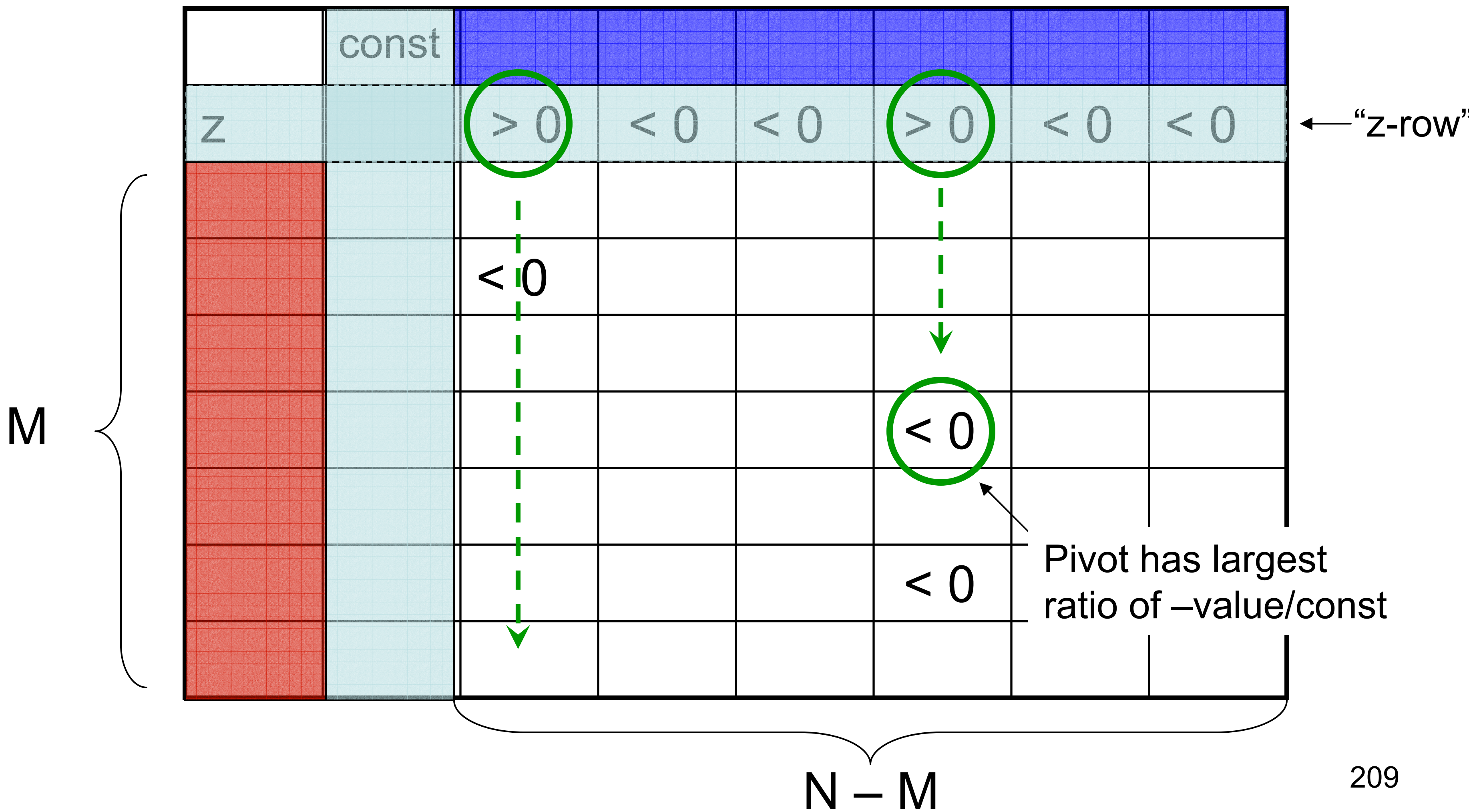
Setting the RH variables to zero, we find that the feasible basic vector we can immediately write is now

$x = (0, +1/3, 0, +9)$, for which $z = 2/3$

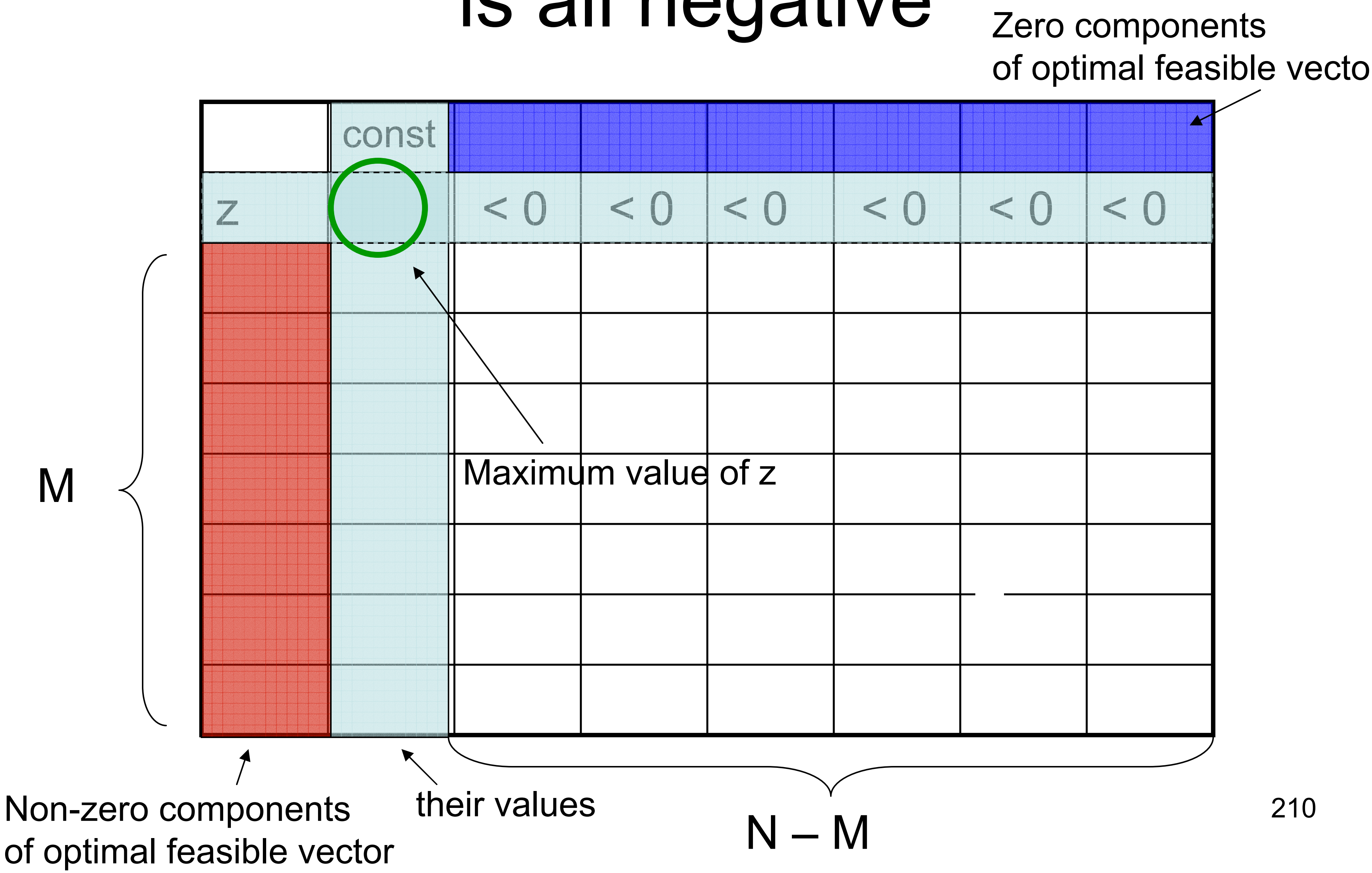
Because all the entries in the z-row are now negative, this is the *optimum* feasible vector, because increasing either RH variable will only decrease z

	const	x_1	x_3
z	+ 2/3	– 1/3	– 11/3
x_2	+ 1/3	– 1/6	+ 1/6
x_4	+ 9	– 1/2	– 7/2

Generalization to N variables and M constraints



Keep pivoting until the “z row” is all negative



Generalization to N variables and M constraints

- The maximum number of required pivots is min(N – M, M)
- The process can (and should) fail if there are no negative entries in a column headed by a positive pivot
  - implies there is no maximum allowed value of z (i.e. no boundary prevents us from going to infinity along that basis vector)

Obtaining the constraints in restricted normal form

- What happens if
  - Some constraints are inequalities
  - Not every constraint equation has a variable that appears in that equation alone
  - Number of constraints $M > N$

- General solution: increase the number of variables!

Treatment of inequalities

Suppose we want

$$\sum_{i=1}^{N} c_{ij}\, x_i \;\leq\; b_j \qquad \text{for one particular } j$$

Introduce new non-negative variable, y_j, (called a "slack variable"), and consider the constraint

$$\sum_{i=1}^{N} c_{ij}\, x_i + y_j = b_j$$

Treatment of inequalities

(Equivalently, for

$$\sum_{i=1}^{N} c_{ij}\, x_i \;\geq\; b_j \qquad \text{for one or more j}$$

we use the constraint

$$\sum_{i=1}^{N} c_{ij}\, x_i - y_j \;=\; b_j \qquad)$$

Treatment of inequalities

- Convert all inequalities according to this prescription
  - This increases the dimensionality of the problem from N to N + K, where K is the number of inequalities
  - Solve by the previous method
  - Ignore the solution for the y_j

Getting the constraints into restricted normal form

Arranging for every constraint equation to have a variable that appears in that equation alone

Change the constraint equations into “zero form”:

$$\text{Define } z_j \equiv \sum_{i=1}^{N} c_{ij}\, x_i - b_j$$

The original M constraints become $z_j = 0$

Getting the constraints into restricted normal form

First use $Z' = -\sum_{j=1}^{M} z_j$ as the "auxiliary" objective function

And adopt the definitions of the z_j as the M constraints. Initially, the z_j are all LH variables and the x_i are all RH variables. This is a problem in restricted normal form

The solution which maximizes Z′ has all the z_j equal to zero* ➔ after the Simplex method is done, all the z_j must become RH variables, and M out of N the x_i will be LH variables.

*if such a solution exists: if not, then the constraints cannot be satisfied simultaneously

Getting the constraints into restricted normal form

But now each of the RH variables appears in exactly 1 constraint equation. We zero out all the z_j, and have a problem involving the x_i in restricted normal form.

This we now solve via the Simplex method, using the original objective function

$$Z = \sum_{i=1}^{N} a_i \, x_i$$

What if the number of constraints exceeds the number of variables?

If $M > N$, we can't possibly have a restricted normal form (there aren't enough variables for each constraint to have a variable that appears in that constraint alone).

But the method described above adds M more variables, so the number of variables always exceeds the number of constraints.