Linear Programming

Greg Plaxton
Theory in Programming Practice, Spring 2004
Department of Computer Science
University of Texas at Austin

Optimization

- In an optimization problem, we are given an objective function, the value of which depends on a number of variables, and we are asked to find a setting for these variables so that
  - Certain constraints are satisfied, i.e., we are required to choose a *feasible* setting for the variables
  - The value of the objective function is maximized (or minimized)
- In general, the objective function and the constraints defining the set of feasible solutions may be arbitrarily complex
- Today we will discuss the important special case in which the objective function and the constraints are linear

Linear Programming

- In linear programming, the goal is to optimize (i.e., maximize or minimize) a linear objective function subject to a set of linear constraints
- Many practical optimization problems can be posed within this framework
- Efficient algorithms are known for solving linear programs
  - Linear programming solving packages routinely solve LP instances with thousands of variables and constraints

Linear Functions

- A linear function of variables $x_1, \ldots x_n$ is any function of the form $c_0 + \sum_{1 \leq j \leq n} c_j x_j$ where
  - The c_j's denote given real numbers
  - The x_j's denote real variables
- Example: $3x_1 - 2x_2 + 10$

Linear Objective Function

- The objective function is the function that we are striving to maximize or minimize
- Suppose our goal is to maximize the linear function $3x_1 - 2x_2 + 10$ (subject to certain constraints that remain to be specified)
- We will get the same result if we drop the constant term and instead simply maximize $3x_1 - 2x_2$
- Also, note that maximizing $3x_1 - 2x_2$ is the same as minimizing $-3x_1 + 2x_2$
- Such a linear objective function is often written in the more compact vector form $c^T x$, where c and x are viewed as $n \times 1$ column vectors, the superscript T denotes transpose, and multiplication corresponds to inner product

Linear Constraints

- A linear constraint requires that a given linear function be at most, at least, or equal to, a specified real constant
  - Examples: $3x_1 - 2x_2 \leq 10$; $3x_1 - 2x_2 \geq 10$; $3x_1 - 2x_2 = 10$
- Note that any such linear constraint can be expressed in terms of upper bound ("at most") constraints
  - The lower bound constraint $3x_1 - 2x_2 \geq 10$ is equivalent to the upper bound constraint $-3x_1 + 2x_2 \leq -10$
  - The equality constraint $3x_1 - 2x_2 = 10$ is equivalent to the upper bound constraints $3x_1 - 2x_2 \leq 10$ and $-3x_1 + 2x_2 \leq -10$

Sets of Linear Constraints: Matrix Notation

- Suppose we are given a set of m upper bound constraints involving the n variables $x_1, \ldots, x_n$
- The constraints can be written in the form $Ax \leq b$ where A denotes an $m \times n$ real matrix, x denotes a column vector of length n (i.e., an $n \times 1$ matrix), and b denotes a column vector of length m
  - The ith inequality, $1 \leq i \leq m$, is $\sum_{1 \leq j \leq n} a_{ij} x_j \leq b_i$
- Similarly, a set of lower bounds constraints may be written as $Ax \geq b$ and a set of equality constraints may be written as $Ax = b$

Nonnegativity Constraints

- The special case of a linear constraint that requires a particular variable, say x_j, to be nonnegative (i.e., $x_j \geq 0$) is sometimes referred to as a nonnegativity constraint
- We will see that such constraints may be handled in a special way within the simplex algorithm, which accounts for their special status
  - Basically, such constraints are handled implicitly rather than explicitly

Standard Form

- A linear programming instance is said to be in *standard form* if it is of the form: Maximize $c^T x$ subject to $Ax \leq b$ and $x \geq 0$
- It is relatively straightforward to transform any given LP instance into this form
  - As noted earlier, a minimization problem can be converted to a maximization problem by negating the objective function
  - We have already seen how to represent lower bound constraints and equality constraints using upper bound constraints
  - If a nonnegativity constraint is missing for some variable x_j (i.e., x_j is "unrestricted"), we can represent x_j by $x'_j - x''_j$ where the variables x'_j and x''_j are required to be nonnegative

Geometric Interpretation

- Suppose we wish to maximize $3x_1 + 2x_2$ subject to the constraints

  (1) $x_1 + x_2 \leq 10$

  (2) $x_1 \leq 8$

  (3) $x_2 \leq 5$

  (4) x_1 and x_2 nonnegative

- Let's draw out the feasible region in the plane

- Now use a geometric approach to find an optimal solution

Simplex Algorithm

- In general, the feasible region defined by constraints of the form $Ax \leq b$ and $x \geq 0$ forms a simplex
- An optimal solution is guaranteed to occur at some "corner" of the simplex
- These "corners" correspond to basic feasible solutions, a notion to be defined a bit later
- The simplex algorithm maintains a bfs, and repeatedly moves to an "adjacent" bfs with a higher value for the objective function
- The simplex algorithm terminates when it reaches a local optimum
  - Fortunately, for the linear programming problem, any such local optimum can be proven to also be a global optimum

Simplex Algorithm: Performance

- In the worst case, the simplex algorithm can take a long time (exponential number of iterations) to converge
  - Fortunately, only pathological inputs lead to such bad behavior; in practice, the running time of the simplex algorithm is quite good
- More sophisticated algorithms (e.g., the ellipsoid algorithm) are known for linear programming that are guaranteed to run in polynomial time

Running Example

- Maximize $4x_1 + 5x_2 + 9x_3 + 11x_4$ ($c^T x$) subject to

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &\leq 15 \\ 7x_1 + 5x_2 + 3x_3 + 2x_4 &\leq 120 \\ 3x_1 + 5x_2 + 10x_3 + 15x_4 &\leq 100 \end{aligned}$$

  ($Ax \leq b$) and $x_j \geq 0$, $1 \leq j \leq 4$

- An application:
  - Variable x_j denotes the amount of product j to produce
  - Value c_j denotes the profit per unit of product j
  - Value a_{ij} denotes the amount of raw material i used to produce each unit of product j
  - Value b_i denotes the available amount of raw material i
  - The objective is to maximize profit

Simplex Example

- Let x_0 denote the value of the objective function and introduce $n = 3$ slack variable x_5, x_6, x_7 to obtain the following equivalent system

- Maximize x_0 subject to the constraints

$$\begin{aligned} x_0 - 4x_1 - 5x_2 - 9x_3 - 11x_4 &= 0 \\ x_1 + x_2 + x_3 + x_4 + x_5 &= 15 \\ 7x_1 + 5x_2 + 3x_3 + 2x_4 + x_6 &= 120 \\ 3x_1 + 5x_2 + 10x_3 + 15x_4 + x_7 &= 100 \end{aligned}$$

- It is easy to see that the above constraints are satisfied by setting x_0 to 0, setting each slack variable to the corresponding RHS (i.e., x_5 to 15, x_6 to 120, and x_7 to 100), and setting the remaining variables to 0
  - Simplex uses this method to obtain an initial feasible solution

Basic Feasible Solution

- There are four constraints in the system given on the preceding slide
- Simplex will consider only solutions with at most four nonzero variables
- Such a solution is called a *basic feasible solution* or *bfs*
- The four variables that are allowed to be nonzero are called the basic variables; the remaining variables are called nonbasic
- We view the initial feasible solution mentioned on the previous slide as a bfs with basic variables $x_0 = 0$, $x_5 = 15$, $x_6 = 120$, and $x_7 = 100$

The High-Level Strategy of the Simplex Algorithm

- The algorithm proceeds iteratively; at each iteration, we have a bfs
- We also have a system of equations, one for each basic variable, such that the associated basic variable has coefficient 1 and the remaining basic variables have coefficient 0
  - In our example, we chose x_0, x_5, x_6, and x_7 as our initial basic variables
  - Note that our initial system of four equations satisfies the aforementioned conditions
- The value of the objective function (i.e., of the variable x_0) increases strictly at each iteration
- When the algorithm terminates, the current bfs is optimal, i.e., x_0 is maximized

A Simplex Iteration

- Check a termination condition to see whether the current bfs is optimal; if so, terminate

- Apply a particular rule to choose an *entering variable*, i.e., a nonbasic variable that will become a basic variable after this iteration

- Apply a second rule to choose a *departing variable*, i.e., a basic variable that will become a nonbasic variable after this iteration

- Perform a *pivot operation* to determine a new bfs, and to update the associated system of equations appropriately

Termination Condition

- Look at the first equality in the set associated with the current bfs
  - In our example, this is

$$x_0 - 4x_1 - 5x_2 - 9x_3 - 11x_4 = 0$$

- In general, the variables in this equation are x_0, with a coefficient of 1, and the nonbasic variables
- If every nonbasic variable has a nonnegative coefficient we terminate; it can be shown that the current bfs is optimal
- In our example, the termination condition does not hold at this point

Selection of the Entering Variable

- Any nonbasic variable with a negative coefficient in the first equation can be chosen as the entering variable
- A good heuristic is to choose the nonbasic variable with the least (i.e., most negative) coefficient
- Such a nonbasic variable is guaranteed to exist since we only select an entering variable when the termination condition does not hold
- In our example, we choose x_4 as the entering variable

Selection of the Departing Variable

- We now increase the entering variable u as much as possible, i.e., until one or more basic variables are driven to zero
- Recall that for each basic variable v, there is exactly one equation E (in the system of equations we are maintaining) in which v has a nonzero coefficient, and in that equation the coefficient of v is 1
- If the coefficient of u is positive in E, then v is a candidate to be the leaving variable since increasing u requires v to be decreased in order to preserve the equality
- If no basic variable is driven to zero by increasing u, we terminate and report that the maximum value of the objective function is unbounded
- Otherwise, for each basic variable v that is driven to zero, we calculate the ratio of the current value of v to the coefficient of u in the equation associated with v to determine how much u needs to increase to drive that v to zero; the minimum ratio determines the departing variable

Back to Our Running Example

- The entering variable is x_4
- By our choice of x_4, x_0 is not a candidate to be a departing variable
- In this example, the remaining basic variables x_5, x_6, and x_7 are all candidates to be the departing variable since the coefficient of x_4 is positive in each of the associated equations
- We calculate the relevant ratios: $\frac{15}{1} = 15$ for x_5, $\frac{120}{2} = 60$ for x_6, and $\frac{100}{15} = \frac{20}{3}$ for x_7
- The minimum ratio is $\frac{20}{3}$, so we select x_7 as the departing variable

Pivot Operation

- Once we have chosen some nonbasic variable u as the entering variable, and some basic variable v as the departing variable, we need to update our bfs and associated system of equations accordingly
- In the new bfs, the (newly) basic variable u is set to the value that causes v to become 0
- The remaining basic variables are updated appropriately
- The departing variable v becomes 0, since we require all nonbasic variables to be zero
- To clarify these computations, and to see how the system of equations is updated, let's look at the first pivot operation applied in our example

Pivot Operation: Example

- Recall that x_4 is the entering variable and x_7 is the departing variable

- Since x_7 is departing, we add appropriate multiples of the fourth equation (the one associated with x_7) to the remaining equations in order to eliminate x_4 from those equations

- We also scale the fourth equation appropriately so that the coefficient of x_4 becomes one

Pivot Operation: Example

- The result is as follows

$$
\begin{aligned}
x_0 - \frac{9}{5}x_1 - \frac{4}{3}x_2 - \frac{5}{3}x_3 + \frac{11}{15}x_7 &= \frac{220}{3} \\
\frac{4}{5}x_1 + \frac{2}{3}x_2 + \frac{1}{3}x_3 + x_5 - \frac{1}{15}x_7 &= \frac{25}{3} \\
\frac{33}{5}x_1 + \frac{13}{3}x_2 + \frac{5}{3}x_3 + x_6 - \frac{2}{15}x_7 &= \frac{320}{3} \\
\frac{1}{5}x_1 + \frac{1}{3}x_2 + \frac{2}{3}x_3 + x_4 + \frac{1}{15}x_7 &= \frac{20}{3}
\end{aligned}
$$

- Interpretation of the RHS's?

Example After Three Iterations

- After three pivot operations, we end up with the basic variables x_0, x_1, x_3, and x_6 and the following associated set of equations

$$
\begin{aligned}
x_0 + \frac{3}{7}x_2 + \frac{11}{7}x_4 + \frac{13}{7}x_5 + \frac{5}{7}x_7 &= \frac{695}{7} \\
x_1 + \frac{5}{7}x_2 - \frac{5}{7}x_4 + \frac{10}{7}x_5 - \frac{1}{7}x_7 &= \frac{50}{7} \\
\frac{2}{7}x_1 + x_3 + \frac{12}{7}x_4 - \frac{3}{7}x_5 + \frac{1}{7}x_7 &= \frac{55}{7} \\
-\frac{6}{7}x_2 + \frac{13}{7}x_4 - \frac{61}{7}x_5 + x_6 + \frac{4}{7}x_7 &= \frac{325}{7}
\end{aligned}
$$

- Since no nonbasic variable has a negative coefficient in the equation associated with basic variable x_0 (the value of the objective function), we terminate

- The maximum value of the objective function is $\frac{695}{7}$ and is attained with $x_1 = \frac{50}{7}$, $x_3 = \frac{55}{7}$, and $x_6 = \frac{325}{7}$

The Dual of a Linear Program

- Suppose we are given a linear program in standard from, i.e., maximize $c^T x$ subject to $Ax \leq b$ and $x \geq 0$
  - When discussing the dual of such an LP, we normally refer to the LP itself as the *primal*
- The *dual* problem is to minimize $y^T b$ subject to $A^T y \geq c$ and $y \geq 0$
  - Note that the dual problem is itself an LP, and can be written in standard form as "maximize $-y^T b = -b^T y$ subject to $(-A)^T y \leq -c$ and $y \geq 0$
  - Note that the dual of the dual is to minimize $z^T(-c) = c^T z$ subject to $(-A)z \geq b$ and $z \geq 0$, which is equivalent to the primal, i.e., the dual of the dual is the primal

Back to the Running Example

- The dual of the example LP given earlier is to minimize $15y_1 + 120y_2 + 100y_3$ subject to

$$
\begin{aligned}
y_1 + 7y_2 + 3y_3 &\geq 4 \\
y_1 + 5y_2 + 5y_3 &\geq 5 \\
y_1 + 3y_2 + 10y_3 &\geq 9 \\
y_1 + 2y_2 + 15y_3 &\geq 11
\end{aligned}
$$

and $y_i \geq 0$, $1 \leq i \leq 3$

Back to the Running Example

- The optimal values of the dual variables are $y_1^* = \frac{13}{7}$, $y_2^* = 0$, and $y_3^* = \frac{5}{7}$
  - These correspond to the coefficients associated with the slack variables x_5, x_6, and x_7 in the final equation associated with x_0 (see the example simplex execution considered earlier)
- The optimal value of the dual objective is $\frac{695}{7}$, which is the same as the optimal value we found earlier for the primal objective

Weak Duality Theorem

- Primal: Maximize $c^T x$ subject to $Ax \leq b$ and $x \geq 0$
- Dual: Minimize $y^T b$ subject to $A^T y \geq c$ and $y \geq 0$
- Suppose that x^* is a feasible solution to the primal, i.e., $Ax^* \leq b$ and $x^* \geq 0$
- Further assume that y^* is a feasible solution to the dual, i.e., $A^T y^* \geq c$ and $y^* \geq 0$
- Then $c^T x^* \leq (y^*)^T b$, i.e., the value of the primal objective is at most that of the dual objective
- Proof: Note that $(A^T y)^T = y^T A$, so $(y^*)^T A \geq c^T$; since x^* and y^* are nonnegative, we conclude that

$$c^T x^* \leq (y^*)^T A x^* \leq (y^*)^T b$$

Weak Duality Theorem: Corollary

- Primal: Maximize $c^T x$ subject to $Ax \leq b$ and $x \geq 0$
- Dual: Minimize $y^T b$ subject to $A^T y \geq c$ and $y \geq 0$
- Suppose that x^* is a feasible solution to the primal, y^* is a feasible solution to the dual, and $c^T x^* = (y^*)^T b$
- Then x^* is an optimal solution to the primal and y^* is an optimal solution to the dual

Strong Duality

- Primal: Maximize $c^T x$ subject to $Ax \leq b$ and $x \geq 0$
- Dual: Minimize $y^T b$ subject to $A^T y \geq c$ and $y \geq 0$
- If there is a finite optimal value for the primal, then the dual has the same finite optimal value
- If the optimal solution to the primal is unbounded, then the dual is infeasible