Simplex Algorithm

In the worst-case this algorithm for LP runs in exponential in the number of variables and constraints $(m+n)$, but in practice it runs often quite fast.

Consider the following LP in standard form:

$$\begin{aligned} \text{Maximize} \quad & 3x_1 + x_2 + 2x_3 \\ \text{s.t.} \quad & x_1 + x_2 + 3x_3 \leq 30 \\ & 2x_1 + 2x_2 + 5x_3 \leq 24 \\ & 4x_1 + x_2 + 2x_3 \leq 36 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

Corresponding slack form:

$$\begin{aligned} z &= 3x_1 + x_2 + 2x_3 \\ x_4 &= 30 - x_1 - x_2 - 3x_3 \\ x_5 &= 24 - 2x_1 - 2x_2 - 5x_3 \\ x_6 &= 36 - 4x_1 - x_2 - 2x_3 \end{aligned}$$

- A solution is feasible if $x_i \geq 0, \forall i = 1, \ldots, 6$.

Basic solution: Set all non-basic variables on the RHS to zero.

Basic sol: $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_6) = (0, 0, 0, 30, 24, 36)$

has objective value

$$z = (3.0) + (1.0) + (2.0) = 0$$

- Simplex algorithm re-writes constraints and objective function so that ~~new~~ basic and non-basic variables are exchanged.

- By above exchange, LP solution does not change.

- A feasible basic solution is almost always maintained by the algorithm

- The goal is to rewrite the LP so that new basic solution has better objective value.

- Select a ~~b~~ non-basic variable x_e whose coefficient c_e in the objective function is positive.

- Increase the value of x_e as much as possible to increase the objective function value.

- In our example x_1 has coeffecient $+3$ in the objective function.

- We cannot increase x_1 arbitrarily since x_4, x_5, x_6 decrease with increasing x_1 and they have to remain positive.

- The constraint

$$x_6 = 36 - 4x_1 - x_2 - 2x_3$$

is the tightest which allows x_1 to increase up to 9.

- Switch the roles of x_6 and x_1

$$x_1 = 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4}$$

Rewrite other constraints with x_6 on the right:

$$x_4 = 30 - x_1 - x_2 - 3x_3$$
$$= 30 - \left(9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4}\right) - x_2 - 3x_3$$
$$= 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4}$$

$$x_5 = \cdots\cdots$$

Similarly, eliminate x_1 from the objective function and bring in x_6. New LP in re-written form:

$$z = 27 + \frac{x_2}{4} + \frac{x_3}{2} - \frac{3x_6}{4}$$
$$x_1 = 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4}$$
$$x_4 = 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4}$$
$$x_5 = 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2}.$$

This entire operation is called <u>pivot</u>.
Pivot chooses a nonbasic variable x_e called <u>entering variable</u> and make it basic replacing a basic variable called <u>leaving variable</u> denoted x_ℓ.

Continue Pivoting:

- Choose x_3: The third constraint

$$x_5 = 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2}$$ is the tightest.

So, $x_e = x_3$, $x_l = x_5$, LP re-written:

$$z = \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - \frac{11x_6}{16}$$

$$x_1 = \frac{33}{4} - \frac{x_2}{16} + \frac{x_5}{8} - \frac{5x_6}{16}$$

$$x_3 = \frac{3}{2} - \frac{3x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8}$$

$$x_4 = \frac{69}{4} + \frac{3x_2}{16} + \frac{5x_5}{8} - \frac{x_6}{16}.$$

Basic soln $= \left(\frac{33}{4}, 0, \frac{3}{2}, \frac{69}{4}, 0, 0\right)$

$$z = \frac{111}{4}.$$

- Choose x_2 (this is the only way to increase objective value)

Three constraints has 132, 4, ∞ as upper bounds.

So, $x_e = x_2$, $x_l = x_3$

$$z = 28 - \frac{x_3}{6} - \frac{x_5}{6} - \frac{2x_6}{3}$$

$$x_1 = 8 + \frac{x_3}{6} + \frac{x_5}{6} - \frac{x_6}{3}$$

$$x_2 = 4 - \frac{8x_3}{3} - \frac{2x_5}{3} + \frac{x_6}{3}$$

$$x_4 = 18 - \frac{x_3}{2} + \frac{x_5}{2}.$$

All coefficients now in the objective function are negative. At this point we have achieved optimal solution.

So, the solution $(8, 4, 0, 18, 0, 0)$ which gives objective value 28 is the optimal solution.

Pivot: takes (N, B, A, b, c, v) in slack form, index l for leaving variable, index e for entering variable.

Input: (N, B, A, b, c, v, l, e)

Output: $(\hat{N}, \hat{B}, \hat{A}, \hat{b}, \hat{c}, \hat{v})$ new slack form.

Pivot algorithm

Pivot (N, B, A, b, c, v, l, e)

* compute coefficients of the equation of x_e

$\hat{b}_e := b_l / a_{le}$

for each $j \in N - \{e\}$

$\hat{a}_{ej} := a_{lj} / a_{le}$;

$\hat{a}_{el} := 1 / a_{le}$;

* compute coefficients of other constraints

for $i \in B - \{l\}$

do $\hat{b}_i := b_i - a_{ie} \hat{b}_e$

for $j \in N - \{e\}$

do $\hat{a}_{ij} := a_{ij} - a_{ie} \hat{a}_{ej}$

$\hat{a}_{il} := - a_{ie} \hat{a}_{el}$;

* compute the objective function.

$\hat{v} := v + c_e \hat{b}_e$;

for each $j \in N - \{e\}$

do $\hat{c}_j := c_j - c_e \hat{a}_{ej}$

$\hat{c}_l := - c_e \hat{a}_{el}$

* compute basic and nonbasic variables

$\hat{N} := N - \{e\} \cup \{l\}$

$\hat{B} := B - \{l\} \cup \{e\}$

return $(\hat{N}, \hat{B}, \hat{A}, \hat{b}, \hat{c}, \hat{v})$.

Simplex Algorithm

Issues:

- How do we determine if LP is feasible?
- What do we do if LP is feasible but initial basic solution is not?
- How do we determine LP is unbounded
- How to choose entering and leaving variable

Initialize (A, b, c): takes an LP in standard form.

$A: \{a_{ij}\}$ $m \times n$ matrix
$b: (b_i)$: m-vector
$c: (c_j)$: n-vector

If LP is infeasible, it returns saying LP is infeasible. Otherwise, it returns a slack form where initial basic solution is feasible.

Simplex (A, b, c): takes LP in standard form returns n-vector $\bar{x} = (\bar{x}_j)$, an optimal solution.

```
Simplex(A, b, c)
  (N, B, A, b, c, v) := Initialize(A, b, c)
  while j ∈ N has c_j > 0
    do choose e ∈ N for which c_e > 0
      for i ∈ B
        do if a_ie > 0
             then Δ_i := b_i / a_ie
              else Δ_i := ∞

        Choose l ∈ B that minimizes Δ_i

        If Δ_l = ∞
          then return "unbounded"

          else
             (N, B, A, b, c, v) := Pivot
                              (N, B, A, b, c, v, l, e)

  for i := 1 to n
        do if i ∈ B
             then x̄_i := b_i
              else x̄_i := 0
  return (x̄_1, x̄_2, ..., x̄_n).
```

Lemma 1. Simplex returns a feasible solution or an "unbounded" solution.

Proof. loop invariant for the outer while loop:
At the beginning of the loop:

1. the slack form is equivalent to the original slack form returned by Initialize
2. for $i \in B$, $b_i \geq 0$
3. the basic solution is feasible.

Show the three invariants at

(a) Initialization, (b) in the middle, (c) at termin

(See the proof in the book for all three cases).

Termination

Now we show that Simplex can always be made to terminate. (why can it cycle?)

$$z = x_1 + x_2 + x_3 \qquad \xrightarrow[x_\ell = x_4]{x_e = x_1} \qquad z = 8 + x_3 - x_4$$
$$x_4 = 8 - x_1 - x_2 \qquad\qquad x_1 = 8 - x_2 - x_4$$
$$x_5 = x_2 - x_3 \qquad\qquad x_5 = x_2 - x_3$$

$$\xrightarrow[x_\ell = x_5]{x_e = x_3} \qquad \left.\begin{aligned} z &= 8 + x_2 - x_4 - x_5 \\ x_1 &= 8 - x_2 - x_4 \\ x_3 &= x_2 - x_5 \end{aligned}\right\} \text{Objective value didn't change}$$

Fortunately, if we pivot with x_2 entering and x_1 leaving, objective value increases. But, it can happen that objective value remains same with successive pivoting. Then, Simplex algorithm "cycles" through identical slack forms.

How can we detect "cycles"?

Lemma 2 Let I be a set of indices. For $i \in I$, α_i, β_i are reals, x_i real variable, γ a real no. If $\sum \alpha_i x_i = \gamma + \sum \beta_i x_i$ then $\alpha_i = \beta_i$ and $\gamma = 0$.

Lemma 3. Let (A, b, c) be an LP in standard form. Given a set B of basic variables, the slack form is uniquely determined.

Proof. $z = v + \sum_{j \in N} c_j x_j$, $x_i = b_i - \sum_{j \in N} a_{ij} x_j$ $i \in B$

$z = v' + \sum_{j = N} c'_j x_j$, $x_i = b'_i - \sum_{j \in N} a'_{ij} x_j$ $i \in B$

be two slack forms for same B.

$$0 = (b_i - b'_i) - \sum (a_{ij} - a'_{ij}) x_j, \quad i \in B$$

$$\sum_{j \in N} a_{ij} x_j = (b_i - b'_i) + \sum_{j \in N} a'_{ij} x_j, \quad i \in B.$$

Apply Lemma 2, to claim

$$b_i = b'_i, \quad a_{ij} = a'_{ij}.$$

Also, show $c = c'$, $v = v'$.

Lemma 4. If Simplex fails to terminate in at most $\binom{n+m}{m}$ iterations, then it cycles.

Proof. There are at most $\binom{n+m}{m}$ different B since $|B| = m$ and total variables is $n+m$. Thus there are at most $\binom{n+m}{m}$ unique slack forms. Conclusion follows.

Cycling can be avoided by choosing entering and leaving variables carefully. Break ties by choosing the variable with the smallest index: Bland's rule.

Lemma 5. If ties are always broken with Bland's rule, Simplex terminates.

We will show that when Simplex returns a feasible solution, it is always optimal. This is shown by duality.