8

LINEAR OPTIMIZATION

8.1 Equality constrained linear programming

In equality constrained linear programming, we aim to solve following optimization problems,

$$
\begin{aligned}
\min_{x\in\mathbb{R}^n} \quad & c_1x_1 + \cdots + c_nx_n \\
\text{subject to} \quad & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\
& a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\
& \vdots \\
& a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m.
\end{aligned}
$$

Writing compactly in matrix form, we have

Definition 8.1.1 (equality constrained linear programming). *A equality constrained linear programming is given as:*

$$
\begin{aligned}
&\min_{x\in\mathbb{R}^n} c^Tx \\
&\text{subject to } Ax = b
\end{aligned}
$$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, $rank(A) = m < n$, and $Ax = b$ is assumed to be consistent.

Remark 8.1.1.

- **Assuming** $rank(A) = m$ **will not cause lose of generality**, since if we can always make A full rank by removing linear dependent rows.
- **The situation** $m \geq n$ **is trivial**: either we have only one feasible solution or we do not have any feasible solution.
- **Assuming** $Ax = b$ **is consistent** will ensure feasible region is not empty.

Under constraint $Ax = b$, the feasible region is a point x_0 satisfying $Ax_0 = b$ plus the null space of A, denoted by $\mathcal{N}(A)$. If c is not perpendicular to $\mathcal{N}(A)$, then the objective function c^Tx can take value from $-\infty$ to ∞. Otherwise, the objective function will take a constant value within the whole feasible region.

We have the following summary.

Theorem 8.1.1 (solution to equality constrained linear programming). *Suppose $rank(A) = m < n$ and $Ax = b$ is consistent. The solution to the equality constrained linear programming problem has exactly two possibilities:*

1. *If $Ax = b$ is consistent and $c \perp \mathcal{N}(A)$, then every solution to $Ax = b$ is a minimizer; moreover, these minimizers are all equal and they are all global minimizers.*
2. *If $Ax = b$ is consistent and $c \not\perp \mathcal{N}(A)$, then the optimization problem is unbounded blow.*

Moreover, check $c \perp \mathcal{N}(A)$ is equivalent to whether $c \in \mathcal{R}(A^T)$ or the consistence of linear equation $c = A^T z$.

Proof. (1) We can decompose every feasible solution $x = x_0 + y$ uniquely, where y is a vector in the null space of $A(y \in \mathcal{N}(A))$, x_0 is a solution to $Ax_0 = b$ and $x_0 \perp \mathcal{N}(A), x_0 \in \mathcal{R}(A^T)$(note that $\mathcal{R}(A^T) \perp \mathcal{N}(A)$ by the fundamental theorem of linear algebra). Then we have minimum $c^T x = c^T x_0 + c^T y = c^T x_0$ because $c \perp \mathcal{N}(A)$, with x_0 being the minimizer. Other feasible solution can be written as $x + w$, where $w \in \mathcal{N}(A)$ and $c^T w = 0$; (2) The objective function can be written as $c^T x = c^T x_0 + c^T y$, which can be made to $-\infty$, since y can be an arbitray vector in the null space of A;

(3) Since $\mathcal{R}(A^T) \perp \mathcal{N}(A), \mathcal{N}(A)^\perp = \mathcal{R}(A^T)$, from fundamental theorem of linear algebra. $\square$

8.2 Inequality constrained linear programming

8.2.1 Linear optimization with inequality constraints

In equality constrained linear programming, we aim to solve following optimization problems,

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & c_1x_1+\cdots+c_nx_n \\ \text{subject to} \quad & a_{11}x_1+a_{12}x_2+\cdots+a_{1n}x_n=b_1 \\ & a_{21}x_1+a_{22}x_2+\cdots+a_{2n}x_n=b_2 \\ & \vdots \\ & a_{m1}x_1+a_{m2}x_2+\cdots+a_{mn}x_n=b_m. \end{aligned}$$

Writing compactly in matrix form, we have

Definition 8.2.1 (canonical form inequality constrained linear optimization). *A linear programming canonical form is:*

$$\begin{aligned} & \textit{minimize } c^Tx \\ & \textit{subject to } Ax \geq b \end{aligned}$$

where $A\in\mathbb{R}^{m\times n}, x\in\mathbb{R}^n$. ***And we require*** $rank(A)=n\leq m$ ***(tall and thin)***

Remark 8.2.1 (generality of canonical form)**.** The canonical form covers both equality and inequality constraints. To see this, consider an equality constraint $\{a_i^Tx=b_i, i=1,...,q\}$, we can convert it to inequality constraints as $\{a_i^Tx\geq b_i, -a_i^Tx\leq -b_i\}$

Remark 8.2.2 (Why we require $rank(A)=n$ and $m\geq n$)**.**

- If $rank(A)<n$, then $Ax=b$ has the solution being an affine set. If c is not perpendicular to the $\mathcal{N}(A)$, then c^Tx will be unbounded(then the minimizer might not exist).
- If $m<n$, then $rank(A)\leq m<n$, we end up with the first situation.

8.2.2 Geometry of linear programming

The feasible set can be viewed the intersection of finite half spaces. When moving in the $-c$ direction in the feasible region, the objective function decreases. There are several critical observations.

- The feasible region is an open space extending to infinity if A is not full column rank. In this case, the objective function is usually unbounded below unless c is perpendicular to $\mathcal{N}(A)$[Figure 8.2.1(a)].
- Under the assumption $rank(A) = n$ and $m \geq n$, the feasible set is either a 'close' polyhedron [Figure 8.2.1(a)(b)]. or a 'half-close-half-open' polyhedron [Figure 8.2.1(d)].
- If the feasible region is a 'close' polyhedron, the minimizer could be found by tracking down along $-c$ until hitting an extreme vertex.
- If the feasible region is a 'half-close-half-open' polyhedron, the objective function could be unbounded below if $-c$ pointing outward.

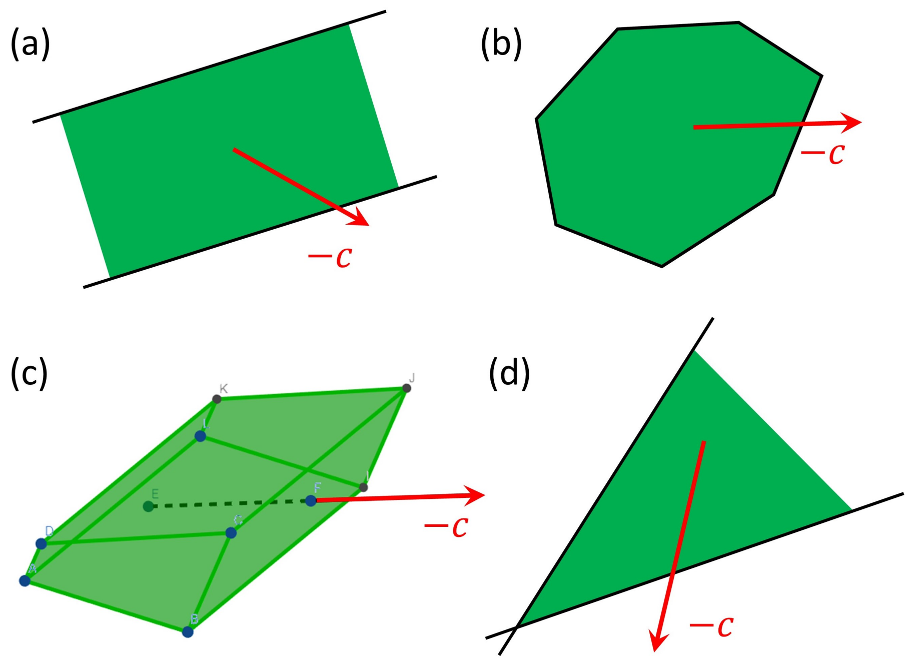

Figure 8.2.1: The geometry of linear programming. (a) The feasible region is an open space extending to infinity if A is not full column rank. (b-d) Example feasible regions if $rank(A) = n, m \geq n$. Red arrows are direction of $-c$. When moving along $-c$ in the feasible region, the objective function will decrease.

Example 8.2.1. Consider a company that produces product A and B. A and B can be sold at prices of \$8 and \$10, respectly. The cost to produce A and B are \$4 and \$7. The production budget is \$100. Let x and y denote the quantity of A and B to produce. Then the goal to maximize profit is given by the following linear program

$$\begin{cases} \max & f(x,y) = 4x + 3y \\ \text{subject to} & 4x + 7y \leq 100 \\ & y \geq 0 \\ & x \geq 0 \end{cases}$$

8.2.3 Optimality property and condition

Lemma 8.2.1 (local minimum implies global minimum). *Let x^* be a local minimizer. Then x^* is also a global minimizer.*

Proof. let x^* be a local minimizer, suppose there is a point $x' \neq x^*$ being a global minimum such that $c^T x' < c^T x^*$. Then

$$c^T(x^* + \alpha(x' - x^*)) = \alpha c^T x^* + (1-\alpha)c^T x' < \alpha c^T x^* + (1-\alpha)c^T x^* < c^T x^*$$

for $\alpha \in [0,1]$. Therefore, starting from $c^T x^*$, if we move along $(x' - x^*)$ for a sufficiently small step, the objective function value will decrease. This contradicts that x^* is local minimizer. □

Theorem 8.2.1 (first-order KKT conditions are sufficient and necessary). *[1, lec 6] If there exists a KKT point, i.e., a point $(x^*, \lambda_a), x^* \in \mathbb{R}^n$ that satisfies*

$$Ax^* \geq b, c = A_a^T y_a, y_a \geq 0,$$

where A_a is the active constraint matrix at x^*, then (x^*, λ_a) is a minimizer.

Moreover, the objective function is unbounded below on the feasible region if and only if there does not exists a $y_a \geq 0$ such that $c = A_a^T y_a$.

Proof. (1) The first order KKT necessary condition directly from Theorem 7.4.2Theorem 7.3.2. We have the kkT condition on (x^*, y^*) given by,

$$\begin{aligned} Ax^* \geq b, x^* \in \mathbb{R}^n (primal feasibility) \\ A^T y^* = c, y^* \geq 0, y^* \in \mathbb{R}^m (dual feasibility) \\ [Ax^* - b]_i [y^*]_i = 0, \forall i = 1, ..., m (complementarity) \end{aligned}$$

When a constraint s is inactive, then $[y^*]_s = 0$, therefore $A^T y^* = c$ can be simplified as $c = A_a^T y_a$. For active constraints, note that the complementary conditions for these active index are automatically satisfied. (2)(sufficiency) Note that since $y^* \geq 0, Ax^* \geq b$, the complementary condition is also equivalent to

$$(Ax^* - b)^T y^* = 0 \Leftrightarrow b^T y^* = c^T x^*.$$

Using this result, we can show the KKT condition is also sufficient: Let $\bar{x}$ be any other feasible point such that $A\bar{x} \geq b$. Then

$$c^T \bar{x} = (A^T y^*)^T \bar{x} = (A\bar{x})^T y^* \geq b^T y^* = c^T x^*,$$

where we used the fact that $A\bar{x} \geq b, y^* \geq 0$. □

Lemma 8.2.2 (uniqueness of minimizer, affine geometry of minimizers). *If the linear optimization problem has a minimizer, then there are* ***exactly*** *two possibilities:*

- *it has a unique minimizer.*
- *it has infinitely many minimizers forming an convex hull; or equivalently, if it has more than one minimizers, then it has infinitely many minimizers.*

Proof. Given two minimizers (x_1, y_1) and (x_2, y_2), then it is easy to see that the convex combination of the two will still satisfy the KKT condition [Theorem 8.2.1]. It can be generalized to multiple minimizers. □

Example 8.2.2. Consider the linear programming problem illustrated in Figure 8.2.2. The red edge that normal to c are all minimizers.

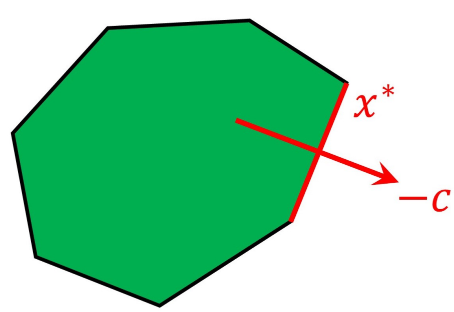

Figure 8.2.2: Demonstration on multiple minimizers forming a convex set.

8.2.4 Standard form of linear programming

Definition 8.2.2 (standard form linear optimization). *[2, p. 4] A linear programming standard form is:*

$$\begin{aligned} &\text{minimize } c^T x \\ &\text{subject to } Ax = b \\ &\qquad\qquad x \geq 0 \end{aligned}$$

where $A \in \mathbb{R}^{m\times n}, x \in \mathbb{R}^n$

Reductions to standard forms from canonical form in two steps:

- For an unrestricted x_i, replaced with $x = x_i^+ - x_i^-$ and add constraint $x_i^+ \geq 0, x_i^- \geq 0$
- For an inequality constraint $a^T x \geq b_i$, replaced with

$$a^T x - s = b$$

$$s \geq 0$$

8.2.5 Application examples

Lemma 8.2.3. *The optimization problem*

$$\min_x [\max_{1\leq j\leq k} \{l_j(x)\}]$$

where $x \in \mathbb{R}^n, l_j(x) = a_j^T x + b_j, a_j \in \mathbb{R}^n, b_j \in \mathbb{R}$ *is equivalent to*

$$\min_{x,v} v, \ s.t. \ l_j(x) \leq v, \forall 1 \leq j \leq k.$$

Proof. Note that the constraints

$$l_j(x) \leq v, \forall 1 \leq j \leq k, \forall x, v,$$

implies

$$v \geq \max_{1\leq j\leq k} l_j(x), \forall v, x$$

Note that left side is a function of v and right side is a function of x. We therefore minimize both sides on v and x simultaneously and we get

$$\min_{v,x} v = \min_x [\max_{1\leq j\leq k} \{l_j(x)\}].$$

□

8.3 Linear programming geometry and simplex algorithm

8.3.1 Geometrical approach to linear programming

8.3.1.1 *Overview*

Following the initial discussion on the geometry of linear programming [subsection 8.2.2], we now aim to develop an algorithm based on the geometry. Given a bounded feasible region [Figure 8.3.1(a)], an intuitive way to search for the extreme vertex that minimize the objective function is to move from one vertex to another along edges that would decrease objective function values. For a bounded feasible region [Figure 8.3.1(b)], we would either find a minimizer or find out if the objective function is unbounded below.

Based on this insight, a complete algorithm should consist the following steps:

- Determine if a vertex is local minimizer or not.
- If a vertex is not the minimizer, determine which edge or direction to move to.
- Determine the step size to move to exactly a new vertex.

These steps are indeed the essence of the famous Simplex algorithm. In the following sections, we will examine how to mathematically formulate the procedures for each step.

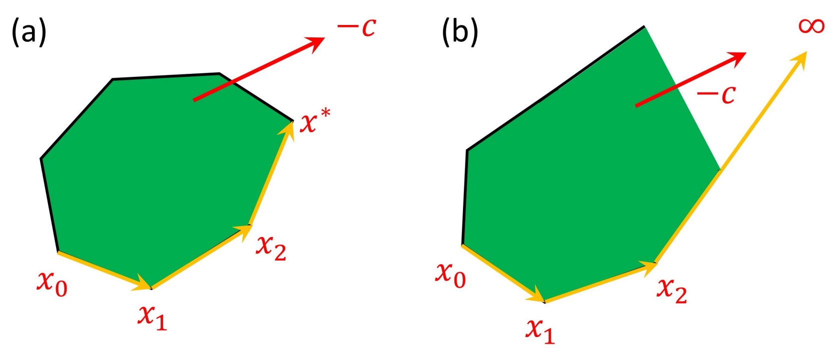

Figure 8.3.1: Overview of geometry approach to linear programming.

8.3.1.2 *Vertex and optimality*

Definition 8.3.1 (vertex, degeneracy). *Given a set of m linear constraints in n variables, a vertex is a feasible point for which the active-constraint matrix contains* ***at least one subset of** n* ***linearly independent rows****, i.e., rank*$(A_a) = n$.

*If **exactly** n constraints are active at a vertex, the vertex is said to be **non-degenerate**. If more than n constraints are active at a vertex, then it is said to be **degenerate**.*

Remark 8.3.1 (interpretation).

- **Any vertex** $x^* \in \mathbb{R}^n$ **is a single point**, since $A_a x^* = b_a$ has an unique solution when $rank(A_a) = n$.
- A vertex can only exist when $m \geq n$. Algebraically, if $m < n$, and $rank(A_a) \leq m < n$. Geometrically, if $m < n$, then $Ax = 0$ has infinite solutions, and solution to $Ax* = b$ will a affine plane instead of a point.

Definition 8.3.2 (adjacent vertex). *Let x_1 and x_2 be two vertices of the feasible region $\mathcal{F}$ for the constraints $Ax \geq b$, and let A_1 and A_2 denote their associated active-constraint matrices. We say that x_1 and x_2 are **adjacent vertices** if the matrix B formed from the rows common to both A_1 and A_2 has $rank(B) = n - 1$.*

*In particular, if x_1 and x_2 are both **non-degenerate adjacent vertices**, then A_1 and A_2 has rank $n - 1$.*

Lemma 8.3.1 (existence of a vertex). *Consider the constraints $Ax \geq b$, where $A \in \mathbb{R}^{m \times n}$. If*

- *there exists at least one feasible point;*
- *$rank(A) = n$*

then a vertex exists.

Proof. Suppose x_0 is a feasible but not a vertex. Let $A_{a(x_0)}$ be the active constraint matrix at x_0. Since x_0 is not vertex, then $dim(\mathcal{N}(A_{a(x_0)})) > 0$. Therefore, we can select $p \in \mathbb{R}^n, p \in \mathcal{N}(A_{a(x_0)})$ such that $x_1 = x_0 + \alpha p, \alpha \in \mathbb{R}$ will still satisfy the original $A_{a(x_0)} x_1 = b_{a(x_0)}$ since $A_{a(x_0)} p = 0$. More importantly, we can select α such that we activate one more constraint. (such α must exist since our feasible region does not contain an affine set(see subsection 8.2.2]).

Therefore, as we continue the process, as long as x_k is not a vertex such that $dim(\mathcal{N}(A_{a(x_k)})) = 0$, we can always move to activate more constraint until we hit a vertex. □

Theorem 8.3.1 (existence of a vertex minimizer). *[1, lec 6] Consider the linear programming [Definition 8.2.1]. If*

- *at least one feasible point exists*
- *$rank(A) = n$*

- *the objective function is bounded blow on the feasible region (or equivalently, the objective function has a minimizer on the feasible region)*

Then, there will ***exist a vertex being the minimizer.***

Proof. Suppose x_0 is a minimizer but not a vertex. Let $A_{a(x_0)}$ be the active constraint matrix at x_0. Since x_0 is not vertex, then $dim(\mathcal{N}(A_{a(x_0)})) > 0$. Therefore, we can select $p \in \mathbb{R}^n, p \in \mathcal{N}(A_{a(x_0)})$ such that $x_1 = x_0 + \alpha p, \alpha \in \mathbb{R}$ will still satisfy the KKT condition Theorem 8.2.1 and will not change of value of the objective function since $c^T p = y_a^T A_a p = 0$. More importantly, we can select α such that we activate one more constraint. (such α must exist since our feasible region does not contain an affine set(see subsection 8.2.2).

Therefore, as we continue the process, as long as x_k is not a vertex such that $dim(\mathcal{N}(A_{a(x_k)})) = 0$, we can always move to activate more constraint until we hit a vertex. □

Remark 8.3.2 (implications on optimization algorithms).

- If the feasible set is not empty, then the vertex existence lemma [Lemma 8.3.1] guarantees that there are vertices in the feasible set.
- Further, the vertex minimizer theorem [Theorem 8.3.1] guarantees that among these feasible vertices, one of them must be a minimizer.
- Therefore, we can either enumerate all vertices either by brute force or by smarter vertices search (simplex algorithm).

Theorem 8.3.2 (uniqueness of minimizer and vertex minimizer, non-degenerate case). *Let x^* be a minimizer with active constraints matrix $A_a \in \mathbb{R}^{n\times n}$, that is*

$$Ax^* \geq b, c = A_a^T y_a^*, y_a^* \geq 0.$$

If

$$c = A_a^T y_a^*$$

and

$$y_a^* > 0 \text{ (strict complementarity)},$$

then x^* is the unique minimizer, and it is also a vertex minimizer.

Proof. (1) From the equivalence between search direction and feasible directions [8.3.1], for a non-zero feasible direction p such that $A_a p > 0$(note that since A_a is nonsingular, $A_a p \neq 0$ if $p \neq 0$). Therefore,

$$c^T(x^* + \alpha p) = c^T x + \alpha y_a^* A p > c^T x^*.$$

(2) Because of the fact that if there exists minimizer, then there must exist vertex minimizer, the unique minimizer must be vertex minimizer. □

Theorem 8.3.3 (non-uniquenss of a non-degenerate vertex minimizer). *Let x^* be a* ***non-degenerate vertex*** *for the canonical linear programming problem. If*

$$c = A_a^T y_a^*, y_a^* \geq 0, \text{ and } [y_a^*]_i = 0 \text{ for at least one index } i,$$

then x^* is a vertex minimizer, but it is not unique.

Proof. Consider a search direction [Lemma 8.3.3] p at x^* such that

$$A_a p = e_i.$$

For $\alpha > 0$ sufficiently small, we have

$$c^T(x^* + \alpha p) = c^T x^* + \alpha c^T p = c^T x^* + \alpha p^T A_a^T y_a^a = c^T x^* + [y_a^*]_i = c^T x^*.$$

Since $x^* + \alpha p$ is feasible, therefore it is also a minimizer. Therefore, x^* is not unique. □

Definition 8.3.3 (working set, working matrix). *At the kth iterate x_k(a vertex). The* ***working set*** *$\mathcal{W}_k$ is a index set such that*

- *$\mathcal{W}_k$ contains* ***exactly*** *n indices, i.e., $\mathcal{W}_k = \{w_1, w_2, ..., w_n\}$.*
- *For every $j \in \mathcal{W}_k$, constraint j is active, i.e. $j \in \mathcal{A}(x_k)$.*

The ***working matrix*** *A_k is the $n \times n$ square* ***non-singular*** *matrix such that each row is active constraint matrix row.*

Remark 8.3.3 (geometry of working matrix and optimality).

- Under non-degeneracy assumption, the working matrix A_k consists of the normals of hyperplanes intersecting at x_k.
- If $y_k \geq 0, y_k \in \mathbb{R}^n$, then x_k is the vertex minimizer(see the following lemma).

Lemma 8.3.2. *Let non-degeneracy assumption hold. Let x_k be the kth iterate (a vertex), and A_k be the working matrix. It follows that if*

$$y_k \geq 0, y_k \in \mathbb{R}^n$$

where y_k is the solution of

$$A_k y_k = c,$$

then x_k is the minimizer and (x_k, y_k) is the KKT point.

Proof. From the optimality condition [Theorem 8.2.1], since x_k is feasible, then $A_k x_k \geq b$ is satisfied. □

8.3.1.3 *Descent direction at a vertex*

Definition 8.3.4 (search direction at a vertex). *Let s be the index such that $[y_k]_s < 0$,, then p_k, solved from $A_k p_k = e_s$, is called* ***search direction***.

Lemma 8.3.3 (properties of search direction). *Let x_k be the kth iterate, A_k be the working matrix, and $p_k, A_k p_k = e_s$ be the search direction, where s is the index $[y_k]_s < 0$. Then, given $\alpha > 0$, we have*

- *$x = x_k + \alpha p_k$, with α sufficiently small, will be still in the feasible region, and* ***inactivate*** *the sth constraint and maintain the rest $n-1$ active constraints active.*
- *moving in p_k direction will* ***decrease the objective value****; that is, $L(x) < L(x_k)$.*
- *If no such s exists, then x_k is optimal.*

Proof. (1)From

$$A_k x = A_k(x_k + \alpha p_k) = b + \alpha e_s \geq b$$

we know sth constraint is inactivated and the rest are still active. (2) $c^T x = c^T(x_k + \alpha p_k) = c^T x_k + \alpha c^T p_k = c^T x_k + \alpha (A_k^T y_k)^T p_k = c^T x_k + [y_k]_s < c^T x_k$. □

Note 8.3.1 (search direction is equivalent to the set of all feasible directions at a non-degenerate vertex)**.**

- Given a **vertex** x_k such that $Ax_k = b$,for $\alpha > 0$ sufficiently small and for all directions p such that $A_k p \geq 0$(note that $A_k \neq A$), the new point

$$x_0 + \alpha p$$

will still be a feasible point($A_k(x_0 + \alpha p) \geq 0$ and $x_0 + \alpha p$ will not hit other inactive constraint for α sufficiently small) . Therefore, the set $\{p : A_k p \geq 0\}$ is a subset of the set of all possible feasible directions.
- The set of all possible feasible directions is also given by $\{p : A_k p \geq 0\}$(see Definition 7.4.5].

8.3.1.4 *Stepping along a descent direction*

Definition 8.3.5 (decreasing constraints in the search direction, blocking constraints). *The set of* ***decreasing constraints*** $\mathcal{D}_k$ *at the point* x_k *is given by*

$$\mathcal{D}_k \triangleq \{j : a_j^T p_k < 0\},$$

where a_j^T *is the jth row of the constraint matrix.*

Remark 8.3.4 (geometry of decreasing constraints). When we move along direction p_k, we might encounter hyperplanes(i.e. other constraints). The set of hyperplanes encountered are $\mathcal{D}_k$.

Moreover, if we do not encounter any hyperplanes, then **moving along direction** p_k **will decrease the objective function to -∞ and remain feasibility**.

Definition 8.3.6 (maximum feasible step). *The maximum feasible step* α_k *along a direction* p_k *is given by*

$$\alpha_k = \min_{j \in \mathcal{D}_k} \sigma_j,$$

where

$$\sigma_j = \frac{a_j^T x_k - b}{-a_j^T p_k}.$$

If $\mathcal{D}_k$ ***is an empty set, then move along direction*** p_k ***will decrease the objective function to -∞ and remain feasibility****.*

Remark 8.3.5 (geometric interpretation).

- The maximum feasible step is the maximum length we can move along direction p_k such that we can still maintain the feasibility of $x_{k+1} = x_k + \alpha_k p_k$.
- If we take the maximum feasible step along p_k, we are moving from one vertex to an adjacent vertex [Definition 8.3.2] in the feasible region.
- If we do not take a step $\alpha > \alpha_k$, then we lose the feasibility $x = x_k + \alpha_k p_k$ because we are violating the first constraint we encounter along p_k.

8.3.2 The simplex algorithm

Algorithm 14: The Simplex algorithm (non-degenerate system)

Input: Initial **vertex** x_0 with associated working set $\mathcal{W}_0$ and working matrix $\mathcal{A}_0$

Set $k = 0$

repeat

Compute the unique Lagrange multiplier estimate y_k from $A_k^T y_k = c$.

if $y_k \geq 0$ **then**

return x_k as the minimizer

end

Choose an index s so that $[y_k]_s < 0$, and compute a search direction p_k such that

$$A_k p_k = e_s$$

Compute the residual vector $r(x) = A_k x_k - b$

Compute the set of decreasing constraints $\mathcal{D}_k\{j : a_j^T p_k < 0\}$ where a_j^T is the jth row of A.

Compute the step lengths to the constraints as

$$\sigma_j = \begin{cases} \frac{r_j(x_k)}{-a_j^T p_k}, if\ j \in \mathcal{D}_k \\ +\infty, otherwise \end{cases}$$

Set the maximum feasible step length as

$$\alpha_k = \min_{1 \leq j \leq m} \sigma_j.$$

if $\alpha_k = +\infty$ **then**

return because the objective function can reach -∞ on the feasible region

end

Choose t as the index of a blocking constraint satisfying $\sigma_t = \alpha_k$.

Set $x_{k+1} = x_k + \alpha_k p_k$, $\mathcal{W}_{k+1} = \mathcal{W}_k - \{w_s\} + \{t\}$, and $A_{k+1} = A_k$ with the sth row replaced by the tth row of A.

Set $k = k + 1$.

until *termination condition satisfied*;

Output: approximate minimizer x_k

8.4 Interior point method

8.4.1 Optimality condition

Definition 8.4.1 (standard form linear optimization). *A linear programming standard form is:*

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} c^T x \\ &\text{subject to } Ax = b \\ &\qquad\qquad x \geq 0 \end{aligned}$$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}, m \leq n, rank(A) = m$, *and* $b \in \mathbb{R}^m$.

Remark 8.4.1 (Why requires $m \leq n$ and $rank(A) = m$)**.**

- Assume consistence of $Ax = b$. If A is not full row rank, then we can always eliminate dependence row.
- If $m \geq n, rank(A) = n$, then there is at most one feasible point, thus making the optimization trivial.

Lemma 8.4.1 (standard form linear programming, optimality condition, recap). *[3, p. 359] The optimality conditions are given by*

$$\begin{aligned} c &= A^T y + z, \\ Ax &= b, \\ x_i \cdot z_i &= 0, i = 1, ..., n \\ x &\geq 0, \\ z &\geq 0 \end{aligned}$$

where $A \in \mathbb{R}^{m\times n}, x \in \mathbb{R}^n, y \in \mathbb{R}^m, z \in \mathbb{R}^n, b \in \mathbb{R}^m$.

Proof. See Theorem 8.2.1. □

Note 8.4.1. Find the optimality condition is equivalent to finding a zero of a **nonlinear** equation $F : \mathbb{R}^{m+n+n} \to \mathbb{R}^{m+n+n}$, given by

$$F(x, y, z) = \begin{bmatrix} z + A^T y - c \\ Ax - b \\ XZ\mathbf{1} \end{bmatrix} \text{subject to } x \geq 0, z \geq 0$$

The Jacobian of F is given by

$$J = \begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix}$$

Remark 8.4.2. Usually a full affine step will violate the bound $(x, z) \geq 0$, so we perform a line search along the Newton direction and define the new iterate as

$$(x, y, z) = (x, y, z) + \alpha(\Delta x, \Delta y, \Delta z)$$

for some α such that bounds not being violated. Often $\alpha \ll 1$ to ensure $(x, z) \geq 0$. Therefore affine step usually cannot make large progress.

Lemma 8.4.2 (Jacobian matrix is non-singular if A is full row rank). *If A is full row rank and $x, z > 0$, the Jacobian matrix*

$$\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix}$$

is nonsingular.

Proof. We want to show that

$$\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = 0,$$

only has solution $\Delta x = \Delta y = \Delta z = 0$. From the 2nd row, we know that $\Delta x \in \mathcal{N}(A)$, and from the first row we know that $\Delta z \in \mathcal{R}(A^T)$. Therefore, $\Delta x^T \Delta z = 0$. From the third row, we have

$$Z\Delta x + X\Delta z = 0 \implies \Delta x = -Z^{-1}X\Delta z \implies 0 = \Delta z^T \Delta x = -\Delta z^T Z^{-1} X \Delta z.$$

Because $Z^{-1}X > 0$, we have $\Delta z = 0$. Then, $Z\Delta x + 0 = 0 \implies \Delta x = 0$. From the first row $A^T \Delta y = 0 \implies \Delta y = 0$ since A^T is full column rank. □

8.4.2 Newton step and perturbed system

Definition 8.4.2 (perturbed optimality conditions). *The perturbed optimality conditions with perturbation parameter $\tau > 0$ are given by*

$$\begin{aligned} c &= A^T y + z, \\ Ax &= b, \\ x_i \cdot z_i &= \tau, i = 1, ..., n \\ x &\geq 0, \\ z &\geq 0 \end{aligned}$$

where $A \in \mathbb{R}^{m\times n}, x \in \mathbb{R}^n, y \in \mathbb{R}^m, z \in \mathbb{R}^n, b \in \mathbb{R}^m$.

Remark 8.4.3 (existence and uniqueness of perturbed solutions).

Remark 8.4.4 (log-barrier interpretation). [3, p. 397] The perturbed optimality condition is equivalent to solve

$$\min_x c^T x - \tau \sum_{i=1}^n \ln x_i, \textit{ subject to } Ax = b, x \geq 0.$$

As $\tau \to 0$, we are approach the original standard linear optimization.

Definition 8.4.3 (Primal-dual feasible region). *The primal-dual **feasible set** $\mathcal{F}$ and **strictly feasible set** $\mathcal{F}^0$ are defined as*

$$\begin{aligned} \mathcal{F} &= \{(x,y,z) : Ax = b, c = A^T y + z, (x,z) \geq 0\} \\ \mathcal{F}^0 &= \{(x,y,z) : Ax = b, c = A^T y + z, (x,z) > 0\} \end{aligned}$$

Remark 8.4.5 (Primal-dual feasible region are almost the optimality condition).

- Note that the **Primal-dual feasible region** is much more restrict than the **primal feasible region**(i.e., $Ax = b, x \geq 0$).
- The Primal-dual feasible region satisfies almost all the optimality condition [Theorem 8.2.1] except for the complementary condition.

Definition 8.4.4 (central path in the primal-dual feasible region). *The central path $\mathcal{C}$ associated with the standard LP is defined as*

$$\mathcal{C} = \{(x_\tau, y_\tau, z_\tau] : \tau > 0\},$$

where (x_τ, y_τ, z_τ) satisfies

$$c = A^T y + z, Ax = b, x_i \cdot z_i = \tau, x_i \geq 0, z_i \geq 0, i = 1, ..., n,$$

*for some value $\tau > 0$. **The central path is always in primal-dual feasible region and bounded away from the boundary of the feasible set.***

Definition 8.4.5 (neighborhood of the central path). *The most common neighborhood are*

$$N_2(\theta) = \{(x, y, z) \in \mathcal{F}^0 : \|XZe - \mu e\|_2 \leq \theta\mu\}(two - norm\ neighborhood)$$
$$N_{-\infty}(\gamma) = \{(x, y, z) \in \mathcal{F}^0 : x_i z_i \geq \gamma\mu, \forall i = 1, ..., n\}(wide\ neighborhood)$$

for some constants $\theta \in [0, 1)$, and $\gamma \in (0, 1)$. Typical values used in practice are $\theta = 0.5, \gamma = 10^{-3}$.

Remark 8.4.6. Those Central neighborhoods are all subsets of strictly feasible set.

Definition 8.4.6 (Newton system for perturbed system). *The Newton system for the perturbed optimality condition is given by*

$$\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = - \begin{bmatrix} A^T y + z - c \\ Ax - b \\ XZ\mathbf{1} - \tau\mathbf{1} \end{bmatrix}$$

Remark 8.4.7 (Centering step vs. affine step). [3, p. 398]

- When $\tau = 1$, the new value of (x, y, z) is **getting closer to the central path, which is always in feasible region and bounded away from the boundary of the feasible set.**

- Centering directions are usually biased strongly toward the interior of the non-negative orthant and make little progress in reducing the duality measure $\mu = x^T z$.
- However, make a centering step will probably set the scene for a substantial reduction in μ in the next iteration.
- When $\tau = 0$, we are taking an affine step, which usually lead to large reduction of μ but might hit the boundary $(x, z \geq 0)$. Usually, if $\tau > 0$, it is possible to take a longer step α along the direction $(\Delta x, \Delta y, \Delta z)$ before violating the bounds $(x, z) \geq 0$.

8.4.3 Algorithms

Algorithm 15: Primal-dual long-step path-following algorithm

Input: Initial estimate $(x_0, y_0, z_0) \in N_{-\infty}(\gamma)$. Choose parameter $\gamma \in (0, 1)$ and $0 < \sigma_{min} < \sigma$

Set $k = 0$

repeat

Compute the duality measure $\mu_k \triangleq (x_k^T z_k)/n$. Choose $\sigma_k \in [0, 1]$ and compute a trial step $(\Delta x, \Delta y, \Delta z)$ satisfying

$$\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ Z_k & 0 & X_k \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = - \begin{bmatrix} A^T y_k + z_k - c \\ A x_k - b \\ X_k Z_k \mathbf{1} - \sigma_k \mu_k \mathbf{1} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ X_k Z_k \mathbf{1} - \sigma_k \mu_k \mathbf{1} \end{bmatrix}$$

Choose α_k as the largest value in $(0, 1]$ such that

$$(x_{k+1}, y_{k+1}, z_{k+1}) = (x_k, y_k, z_k) + \alpha_k (\Delta x, \Delta y, \Delta z).$$

such that
$(x_{k+1}, y_{k+1}, z_{k+1}) \in N_{-\infty}(\gamma) = \{(x, y, z) \in \mathcal{F}^0 : x_i z_i \geq \gamma \mu, \forall i = 1, ..., n\}$.

Set $k = k + 1$.

until *termination condition satisfied*;

Output: approximate minimizer x_k

Remark 8.4.8 (interpretation).

- Duality measure $\mu \triangleq x^T z/n$ is not the duality gap, it simply measure how far we are away from optimality $\mu_{opt} = 0$.
- We can terminate the algorithm when μ is sufficiently small.

Remark 8.4.9 (convergence). [3, p. 406]

- It can be showed that the dual measure μ_k is decreasing "sufficiently" for each step therefore as $k \to \infty$, $\mu_k \to 0$ and (x_k, y_k, z_k) converge to the optimal solution.
- The convergence speed is linear.

Remark 8.4.10 (other algorithms). See [3, p. 406] for more algorithms.

8.5 Notes on bibliography

The major references are [1] [4] [2].

For interior point method, see [3].

BIBLIOGRAPHY

1. Robinson, D. *Nonlinear optimization II lecture notes* (Johns Hopkins University, 2015).
2. Bertsimas, D. & Tsitsiklis, J. N. *Introduction to linear optimization* (Athena Scientific Belmont, MA, 1997).
3. Nocedal, J. & Wright, S. *Numerical optimization* (Springer Science & Business Media, 2006).
4. Luenberger, D. & Ye, Y. *Linear and Nonlinear Programming* ISBN: 9783319188423 (Springer International Publishing, 2015).