Linear Programming (LP)

These lecture notes borrow heavily from Chapter 1 of Bertsimas & Tsitsiklis, Introduction to Linear Optimization. Please refer to it for further details, many more examples, and exercises.

Let us start with an example of a LP problem:

$$
\begin{array}{llrl}
\min & 2x_1 - x_2 + x_3 & & \\
\text{subject to} & x_1 + x_2 \quad\quad + x_4 & \leq & 2 \\
 & 3x_2 - x_3 & = & 5 \\
 & x_3 + x_4 & \geq & 3 \\
 & x_1 & \geq & 0
\end{array}
$$

We seek a minimum over all possible $x_1, x_2, x_3, x_4 \in \mathbb{R}$ which satisfy the specified constraints.

So, what makes this an LP?
First, it is an optimisation problem, either a minimisation or maximisation.
Second, the objective function, the one being optimised, is a linear function of the variables.
Lastly, all the constraints are linear functions of the variables.

From now on, we will use vector notation to write things more compactly. Vectors are column vectors unless otherwise specified.

In general, there will be some number n of variables that we optimise over, denoted $x_1, x_2, \ldots, x_n$. These are called the decision variables.

We represent them by the vector $\underline{x} \in \mathbb{R}^n$, where $\underline{x} = (x_1, x_2, \ldots, x_n)^T$ - superscript T denotes transpose.

The objective function has to be linear, so it takes the form $\underline{c}^T \underline{x} = \sum_{i=1}^{n} c_i x_i$, for some constants c_i.

The constraints are also linear, but could be equality or inequality constraints. So we see that the general form of an LP is:

$$
\begin{array}{ll}
\min & \underline{c}^T \underline{x}, \quad \underline{x} \in \mathbb{R}^n \\
\text{subject to} & \underline{a}_i^T \underline{x} \le b_i, \quad i = 1, \ldots, m_1 \\
& \underline{a}_i^T \underline{x} \ge b_i, \quad i = m_1+1, \ldots, m_2 \\
& \underline{a}_i^T \underline{x} = b_i, \quad i = m_2+1, \ldots, m_3
\end{array}
$$

Note that these can be expressed even more compactly in the following form, which is called the Standard Form with Inequality Constraints:

$$
\begin{array}{ll}
\min & \underline{c}^T \underline{x} \quad \text{over } \underline{x} \in \mathbb{R}^n \\
\text{subject to} & A\underline{x} \ge \underline{b}
\end{array}
$$

Here $A \in \mathbb{R}^{m \times n}$ is a matrix, $\underline{b} \in \mathbb{R}^m$, & the ineq. $A\underline{x} \ge \underline{b}$ is to be interpreted componentwise.

Standard Form with Equality Constraints

In fact, the LP can be rewritten in the form

$$\min \ \underline{c}^T \underline{x} \qquad , \ \underline{x} \in \mathbb{R}^n$$

$$\text{subject to} \quad A\underline{x} = \underline{b} \quad , \ A \in \mathbb{R}^{m \times n}, \ \underline{b} \in \mathbb{R}^m$$

$$\underline{x} \geq 0$$

(The dimensions n & m can change when converting a problem with ineq. constraints into one with eq. & positivity constraints).

We'll see how to do this in lectures, and also several examples.

Here's an example that isn't obvious!

$$\min \sum_{i=1}^{n} c_i |x_i| \qquad \text{subj. to} \quad A\underline{x} \geq \underline{b}$$

Now, the function $|x|$ isn't linear, but this can in fact be turned into an LP.

Hint: Observe that $|x_i|$ is the smallest number z_i such that $x_i \leq z_i$ and $-x_i \leq z_i$

$10n^3 = 10^7 \times 1000$ yields $n = 1000$, indicating that the polynomial time algorithm allows us to solve much larger problems.

The point of view emerging from the above discussion is that, as a first cut, it is useful to juxtapose polynomial and exponential time algorithms, the former being viewed as relatively fast and efficient, and the latter as relatively slow. This point of view is justified in many – but not all – contexts and we will be returning to it later in this book.

1.7 Exercises

Exercise 1.1* Suppose that a function $f : \Re^n \mapsto \Re$ is both concave and convex. Prove that f is an affine function.

Exercise 1.2 Suppose that $f_1, \ldots, f_m$ are convex functions from $\Re^n$ into $\Re$ and let $f(\mathbf{x}) = \sum_{i=1}^m f_i(\mathbf{x})$.

(a) Show that if each f_i is convex, so is f.

(b) Show that if each f_i is piecewise linear and convex, so is f.

Exercise 1.3 Consider the problem of minimizing a cost function of the form $\mathbf{c}'\mathbf{x} + f(\mathbf{d}'\mathbf{x})$, subject to the linear constraints $\mathbf{Ax} \geq \mathbf{b}$. Here, $\mathbf{d}$ is a given vector and the function $f : \Re \mapsto \Re$ is as specified in Figure 1.8. Provide a linear programming formulation of this problem.

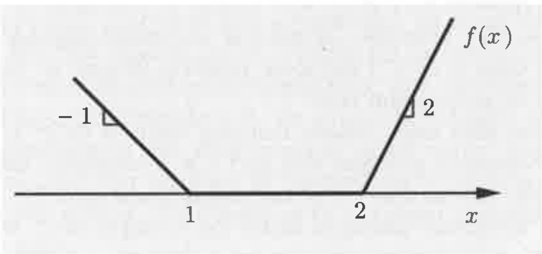


Figure 1.8: The function f of Exercise 1.3.

Exercise 1.4 Consider the problem

$$\begin{array}{ll} \text{minimize} & 2x_1 + 3|x_2 - 10| \\ \text{subject to} & |x_1 + 2| + |x_2| \leq 5, \end{array}$$

and reformulate it as a linear programming problem.

Exercise 1.5 Consider a linear optimization problem, with absolute values, of the following form:

$$\begin{array}{lll} \text{minimize} & \mathbf{c}'\mathbf{x} + \mathbf{d}'\mathbf{y} & \\ \text{subject to} & \mathbf{Ax} + \mathbf{By} \leq \mathbf{b} & \\ & y_i = |x_i|, & \forall\, i. \end{array}$$

Assume that all entries of $\mathbf{B}$ and $\mathbf{d}$ are nonnegative.

(a) Provide two different linear programming formulations, along the lines discussed in Section 1.3.

(b) Show that the original problem and the two reformulations are equivalent in the sense that either all three are infeasible, or all three have the same optimal cost.

(c) Provide an example to show that if $\mathbf{B}$ has negative entries, the problem may have a local minimum that is not a global minimum. (It will be seen in Chapter 2 that this is never the case in linear programming problems. Hence, in the presence of such negative entries, a linear programming reformulation is implausible.)

Exercise 1.6 Provide linear programming formulations of the two variants of the rocket control problem discussed at the end of Section 1.3.

Exercise 1.7 (The moment problem) Suppose that Z is a random variable taking values in the set $0, 1, \ldots, K$, with probabilities $p_0, p_1, \ldots, p_K$, respectively. We are given the values of the first two moments $E[Z] = \sum_{k=0}^K kp_k$ and $E[Z^2] = \sum_{k=0}^K k^2 p_k$ of Z and we would like to obtain upper and lower bounds on the value of the fourth moment $E[Z^4] = \sum_{k=0}^K k^4 p_k$ of Z. Show how linear programming can be used to approach this problem.

Exercise 1.8 (Road lighting) Consider a road divided into n segments that is illuminated by m lamps. Let p_j be the power of the jth lamp. The illumination I_i of the ith segment is assumed to be $\sum_{j=1}^m a_{ij} p_j$, where a_{ij} are known coefficients. Let I_i^* be the desired illumination of road i.

We are interested in choosing the lamp powers p_j so that the illuminations I_i are close to the desired illuminations I_i^*. Provide a reasonable linear programming formulation of this problem. Note that the wording of the problem is loose and there is more than one possible formulation.

Exercise 1.9 Consider a school district with I neighborhoods, J schools, and G grades at each school. Each school j has a capacity of C_{jg} for grade g. In each neighborhood i, the student population of grade i is S_{ig}. Finally, the distance of school j from neighborhood i is d_{ij}. Formulate a linear programming problem whose objective is to assign all students to schools, while minimizing the total distance traveled by all students. (You may ignore the fact that numbers of students must be integer.)

Exercise 1.10 (Production and inventory planning) A company must deliver d_i units of its product at the end of the ith month. Material produced during a month can be delivered either at the end of the same month or can be stored as inventory and delivered at the end of a subsequent month; however, there is a storage cost of c_1 dollars per month for each unit of product held in inventory. The year begins with zero inventory. If the company produces x_i units in month i and x_{i+1} units in month $i+1$, it incurs a cost of $c_2|x_{i+1} - x_i|$ dollars, reflecting the cost of switching to a new production level. Formulate a linear programming problem whose objective is to minimize the total cost of the production and inventory schedule over a period of twelve months. Assume that inventory left at the end of the year has no value and does not incur any storage costs.

Exercise 1.11 (Optimal currency conversion) Suppose that there are N available currencies, and assume that one unit of currency i can be exchanged for r_{ij} units of currency j. (Naturally, we assume that $r_{ij} > 0$.) There also certain regulations that impose a limit u_i on the total amount of currency i that can be exchanged on any given day. Suppose that we start with B units of currency 1 and that we would like to maximize the number of units of currency N that we end up with at the end of the day, through a sequence of currency transactions. Provide a linear programming formulation of this problem. Assume that for any sequence $i_1, \ldots, i_k$ of currencies, we have $r_{i_1 i_2} r_{i_2 i_3} \cdots r_{i_{k-1} i_k} r_{i_k i_1} \leq 1$, which means that wealth cannot be multiplied by going through a cycle of currencies.

Exercise 1.12 (Chebychev center) Consider a set P described by linear inequality constraints, that is, $P = \{\mathbf{x} \in \Re^n \mid \mathbf{a}_i'\mathbf{x} \leq b_i,\ i = 1, \ldots, m\}$. A ball with center $\mathbf{y}$ and radius r is defined as the set of all points within (Euclidean) distance r from $\mathbf{y}$. We are interested in finding a ball with the largest possible radius, which is entirely contained within the set P. (The center of such a ball is called the *Chebychev center* of P.) Provide a linear programming formulation of this problem.

Exercise 1.13 (Linear fractional programming) Consider the problem

$$\begin{array}{ll} \text{minimize} & \dfrac{\mathbf{c}'\mathbf{x} + d}{\mathbf{f}'\mathbf{x} + g} \\ \text{subject to} & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{f}'\mathbf{x} + g > 0. \end{array}$$

Suppose that we have some prior knowledge that the optimal cost belongs to an interval $[K, L]$. Provide a procedure, that uses linear programming as a subroutine, and that allows us to compute the optimal cost within any desired accuracy. *Hint:* Consider the problem of deciding whether the optimal cost is less than or equal to a certain number.

Exercise 1.14 A company produces and sells two different products. The demand for each product is unlimited, but the company is constrained by cash availability and machine capacity.

Each unit of the first and second product requires 3 and 4 machine hours, respectively. There are 20,000 machine hours available in the current production period. The production costs are \$3 and \$2 per unit of the first and second product, respectively. The selling prices of the first and second product are \$6 and \$5.40 per unit, respectively. The available cash is \$4,000; furthermore, 45% of the sales revenues from the first product and 30% of the sales revenues from the second product will be made available to finance operations during the current period.

(a) Formulate a linear programming problem that aims at maximizing net income subject to the cash availability and machine capacity limitations.

(b) Solve the problem graphically to obtain an optimal solution.

(c) Suppose that the company could increase its available machine hours by 2,000, after spending \$400 for certain repairs. Should the investment be made?

Exercise 1.15 A company produces two kinds of products. A product of the first type requires 1/4 hours of assembly labor, 1/8 hours of testing, and \$1.2 worth of raw materials. A product of the second type requires 1/3 hours of assembly, 1/3 hours of testing, and \$0.9 worth of raw materials. Given the current personnel of the company, there can be at most 90 hours of assembly labor and 80 hours of testing, each day. Products of the first and second type have a market value of \$9 and \$8, respectively.

(a) Formulate a linear programming problem that can be used to maximize the daily profit of the company.

(b) Consider the following two modifications to the original problem:

(i) Suppose that up to 50 hours of overtime assembly labor can be scheduled, at a cost of \$7 per hour.

(ii) Suppose that the raw material supplier provides a 10% discount if the daily bill is above \$300.

Which of the above two elements can be easily incorporated into the linear programming formulation and how? If one or both are not easy to incorporate, indicate how you might nevertheless solve the problem.

Exercise 1.16 A manager of an oil refinery has 8 million barrels of crude oil A and 5 million barrels of crude oil B allocated for production during the coming month. These resources can be used to make either gasoline, which sells for \$38 per barrel, or home heating oil, which sells for \$33 per barrel. There are three production processes with the following characteristics:

	Process 1	Process 2	Process 3
Input crude A	3	1	5
Input crude B	5	1	3
Output gasoline	4	1	3
Output heating oil	3	1	4
Cost	\$51	\$11	\$40

All quantities are in barrels. For example, with the first process, 3 barrels of crude A and 5 barrels of crude B are used to produce 4 barrels of gasoline and

Geometry of LP

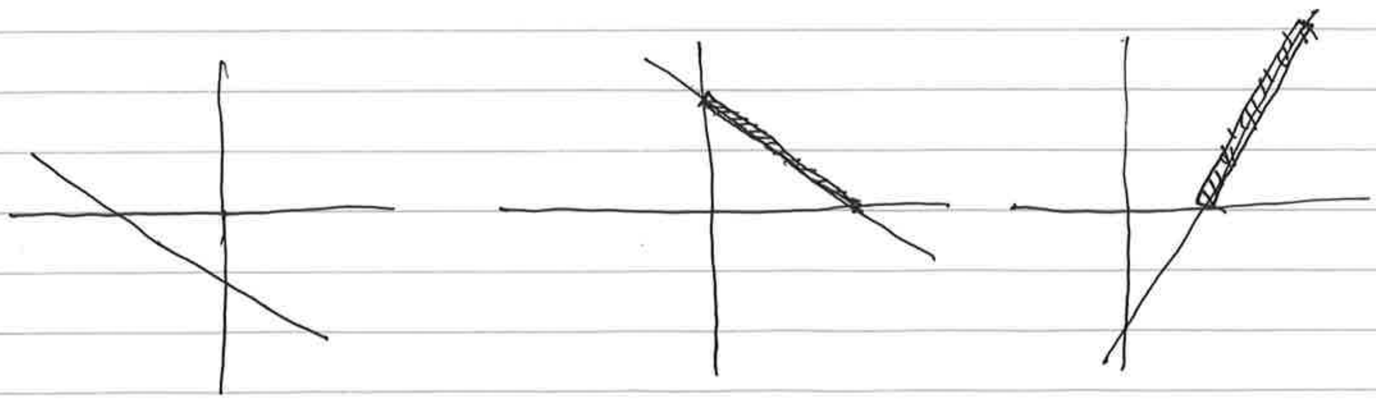
Consider the standard form LP:

$$\min \quad \underline{c}^T \underline{x}, \quad \underline{x} \in \mathbb{R}^n$$
$$\text{subject to} \quad A\underline{x} = \underline{b}, \quad A \in \mathbb{R}^{m \times n}, \ \underline{b} \in \mathbb{R}^m$$
$$\underline{x} \geq \underline{0}$$

Now, the set $\{\underline{x} : \underline{a}_i^T \underline{x} = b_i\}$ is an $(n-1)$-dimensional hyperplane for each i, orthogonal to the vector $\underline{a}_i$.

The set $\{\underline{x} : A\underline{x} = \underline{b}\}$ is the intersection of m such hyperplanes, and hence is an $(n-m)$-dimensional hyperplane, if the constraints are linearly independent.

The feasible set, namely the set of $\underline{x}$ that satisfies all the constraints $A\underline{x} = \underline{b}$ and $\underline{x} \geq 0$, is the portion of this hyperplane that lies in the positive orthant.
There are 3 possibilities: it is empty, it is bounded, or it is unbounded, as shown



(a) empty (b) bounded (c) unbounded

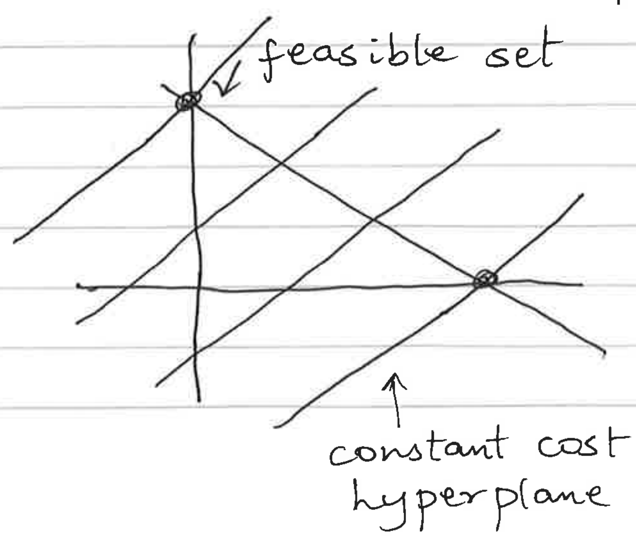
By convention, the minimum over an empty set is defined to be $+\infty$ & the maximum over an empty set to be $-\infty$.

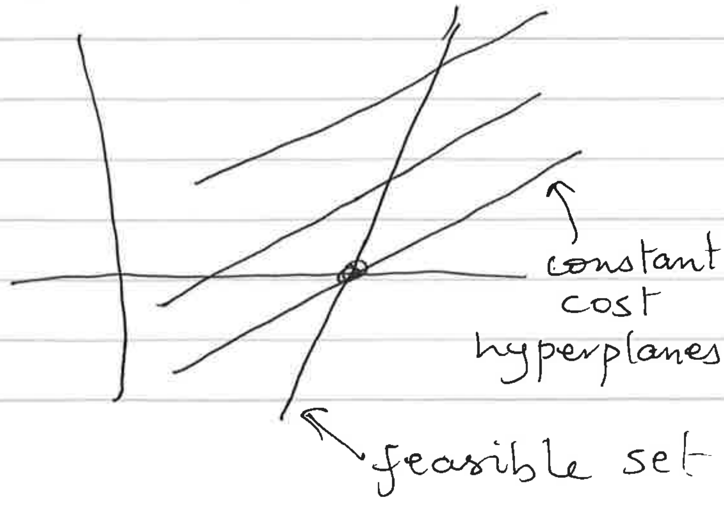
Thus, if an LP in standard form is infeasible (there is no $\underline{x} \in \mathbb{R}^n$ satisfying all the constraints), then the value of the LP is $+\infty$.

If the feasible set is bounded, as in fig (b), then the value (minimum) is finite.

If the feasible set is unbounded, as in fig (c), the value could be finite or $-\infty$.
To determine which, let us look at the cost function, $\underline{c}^T \underline{x}$.

For any K, the set $\{\underline{x} : \underline{c}^T \underline{x} = K\}$ is a hyperplane of $(n-1)$ dimensions; it passes through the origin if $K = 0$, & is orthogonal to the vector $\underline{c}$. Varying K corresponds to parallel translations of this hyperplane. This suggests (and it is true) that the minimum value of the LP is always attained at a corner (extreme point) of the feasible set.





Let P denote the feasible set, i.e., the set of $\underline{x} \in \mathbb{R}^n$ satisfying all the constraints:

$$P = \{ \underline{x} \in \mathbb{R}^n : \underline{x} \geq 0, \; A\underline{x} = \underline{b} \}$$

Suppose P is non-empty. Then, every $\underline{x} \in P$ is called a feasible solution.

We noted above that the optimum (if it is finite) is attained at a corner of P. We now discuss how to find the corners.

The eqns. $A\underline{x} = \underline{b}$ correspond to m equations in n variables. If $m \leq n$ & the eqns. are linearly independent, then they specify an $(n-m)$-dimensional hyperplane in $\mathbb{R}^n$. The feasible set is the intersection of this hyperplane with the positive orthant $\{ \underline{x} \in \mathbb{R}^n : \underline{x} \geqslant 0 \}$.

If we set $n-m$ of the n co-ordinates x_i to 0, we get an additional $n-m$ equations. Together with the m eqns. $A\underline{x} = \underline{b}$, we have n eqns. in variables, which can be solved for $\underline{x}$. Each such solution gives one corner of P, and is called a <u>basic feasible solution</u>.

The simplex algorithm starts from some basic feasible solution, repeatedly finds a direction of improvement, & moves along it to a different basic feasible solution. It repeats this until no such direction can be found. The corresponding corner is clearly a local optimum. In fact, it is also a global optimum, i.e., solves the LP problem.