# 1 Linear Programming

Now, we will talk a little bit about Linear Programming. We say that a problem is an instance of linear programming when it can be effectively expressed in the linear programming framework. A linear programming problem is an *optimization* problem where the optimization function is a linear function.

The term Linear Programming was first used by the miliatary in the 1950s, and so the word programming does not refer to a computer program (code) but rather a schedule or plan. The word linear refers to the fact that the constraints on the schedule or plan are linear inequalities and that the objective function that we are trying to maximize is a linear function.

We are interested in linear programs, because they are very general and can be solved efficiently. As you will see, there is a wide variety of problems that can be formulated as linear programs. In practice, if you find a situation that can be formulated as a linear program with fewer than several thousand variables (and even more in some cases), you will likely be able to solve it quite quickly using existing linear programming packages.

## 1.1 Optimizion problems and convex polytopes

**Definition 1.1** (Optimization Problem)**.** An optimization problem is a function $f : \Sigma \to \mathbb{R}$ along with a subset $F \subseteq \Sigma$. The goal of the problem is to find the $x \in F$ such that for all $y \in F$, $f(x) \leq f(y)$. We often call $F$ the *feasible region* of the problem.

In the case of linear programming, we take $\Sigma = \mathbb{R}^n$, require $f$ to be linear i.e. $f(x) = c^T x$ for some $c \in \mathbb{R}^n$, and the feasible region $F$ is a convex polytope sitting in $n$-dimensional space. Another way of thinking of a convex polytope is that it is the intersection of finitely many half-planes or as the set of points that satisfy a finite number of affine inequalites.

**Definition 1.2** (Convex Polytope)**.** The following are equivalent:

1. For $u_1, \ldots, u_m \in \mathbb{R}^n$ and $b_1, \ldots, b_m \in \mathbb{R}$, the set of $x \in \mathbb{R}^n$ s.t. $u_i^T x \leq b_i$ is a convex polytope.

2. Given a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $b \in \mathbb{R}^m$, the set of $x \in \mathbb{R}^n$ s.t. $Ax \leq b$ is a convex polytope.

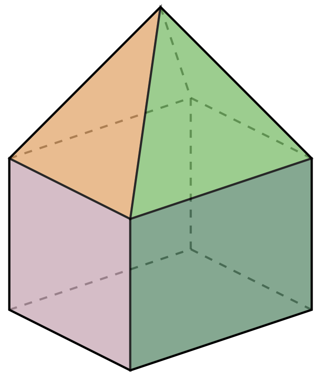

Figure 1.1: An example of a convex polytope. We can consider each face of the polytope as an affine inequality and then the polytope is all the points that satisfy each inequality. Notice that an affine inequality defines a half-plane and therefore is also the intersection of the half-planes.

3. Given a set of points $y_1, \dots, y_k \in \mathbb{R}^n$, the convex hull $\mathbf{conv}(y_1, \dots, y_k)$ is a convex polytope. A convex hull $\mathbf{conv}(y_1, \dots, y_k)$ is the intersection of all convex sets containing $y_1, \dots, y_k$.

## 1.2 Optimums in the Feasible Region

**Definition 1.3** (Feasibility)**.** For a linear program in the standard form of (**??**), we say the *feasible region* is the set $F = \{x : Ax \leq b\}$. If $F \neq \emptyset$, we say $F$ is *feasbile*. We say that the linear program is unbounded if $F$ is unbounded.

A first question we need to ask ourselves when solving a linear program is whether a linear program is feasible. This itself is non-trivial. Assuming, however, that $F$ is feasible, how would we go about finding an optimum? We exploit the two properties that the objective function $c^T x$ is linear and that $F$ is a convex polytope.

**Lemma 1.4** (Local optimums are global optimums)**.** *Given $F \neq \emptyset$, if a point $x \in F$ is a local optimum then it is a global optimum.*

*Proof.* Assume for contradiction, $x$ isn't a global optimum. Then $\exists y \in F$ s.t. $c^T x < c^T y$. But the line $\overline{xy} \in F$. Then for any $\lambda \in (0, 1]$,

$$c^T(\lambda x + (1 - \lambda)y) = \lambda c^T x + (1 - \lambda)c^T y > c^T x \tag{1.1}$$

Meaning, $x$ is not a local optimum as moving towards $y$ increases the objective function. $\square$

**Definition 1.5** (Vertex of a Polytope)**.** Recall that we say $z$ is on the line segement $\overline{xy}$ if there $\exists\lambda \in [0,1]$ s.t. $z = \lambda x + (1-\lambda)y$. A point $z \in F$ is a vertex of the polytope $F$ if it is on no proper line segment contained in $F$.[1]

**Remark 1.6.** *If $v_1, \ldots, v_k$ are the vertices of a polytope $F$, then $F = \boldsymbol{conv}(v_1, \ldots, v_k)$. i.e. $F$ is the convex hull of the vertices.*[2]

**Theorem 1.7.** *Let* **OPT** *be the optimal value of a standard form linear program and assume* **OPT** *is finite. Then* **OPT** *is achieved at a vertex.*

*Proof.* Let $v_1, \ldots, v_k$ be the vertices of $F$. Then every point $x \in F$ can be expressed as $\sum_{i=1}^k \lambda_i v_i$ with each $\lambda_i \geq 0$ and $\sum \lambda_i = 1$. By linearity of the objective function, if $\mathbf{OPT} = c^T x$ for $x \in \mathbb{R}^n$, then necessarily $c^T v_i \geq \mathbf{OPT}$ for some $v_i$, so the optimal value is achieved at the vertex. $\square$

This statement is rather strong in that it means that we only need to check the vertices of a polytope in order to find the optimum. In particular, it also demonstrates that the proper way of thinking about a polytope is not by the number of linear equations that define it, but rather by the number of vertices. This is actually a very fruitful section of algorithm theory. There has been considerable work done to consider how to reduce the number of vertices. However, this is not always optimal. Consider the $n$-dimensional sube. It has $2^n$ vertices. In general if the polytope is defined by $m$ half planes there are up $\binom{m+n}{n}$ vertices.

[1] A vertex is just the $n$-dimensional analog of what you think of as the vertex of a polytope in low dimension.

[2] This can be shown by induction on the dimension. The proof is a little tedious, so it has been ommitted.

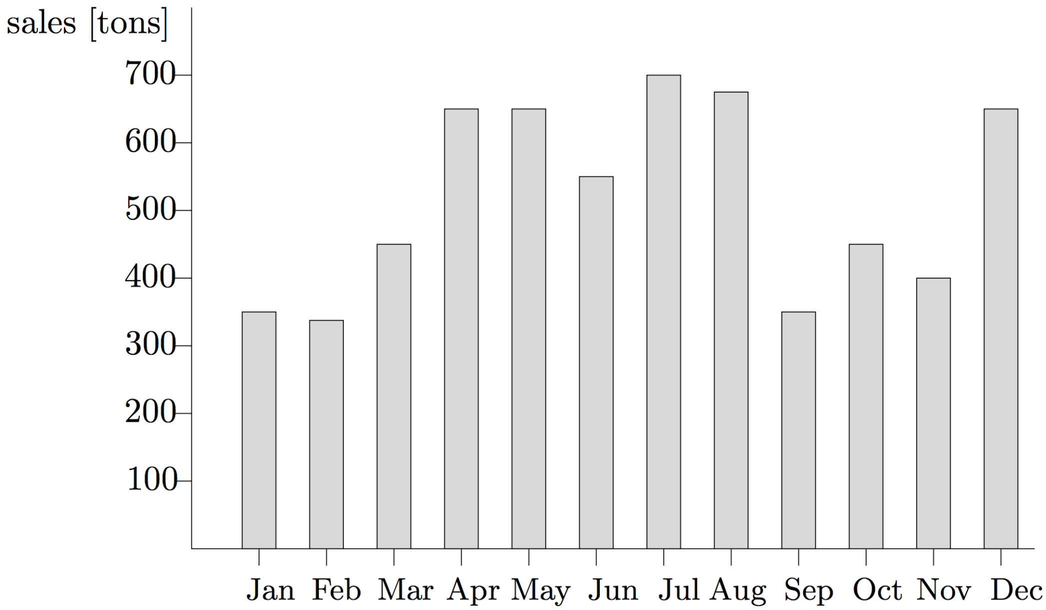


Figure 2.1: Demand schedule for the ice cream company

## 2 An example: Ice cream

I have adapted/borrowed this example from the book Understanding and Using Linear Programming by Jiri Matousek and Bernd Gartner. The problem concerns an ice cream company, which must determine how much ice cream to produce in order to maximize their profits. The company faces an expected seasonal demand given by the chart above.

We assume that company will always choose to meet demand at each month and have no surplus at the end of the year. We also assume that no ice cream is in storage at the beginning of the year, and that they have no production in month 0 (the previous month). This means that the company does not really need to worry about the amount of ice cream they will produce in their model, they just need to figure out when to produce it. Then, to maximize profit in this model we simply need to minimize two costs, $c_1$ which is the cost of storing one ton of ice cream for one month, and $c_2$ which is the cost of changing production by one ton from one month to the next. These costs are quite realistic - storing ice cream we need a freezer is expensive, and it is also expensive to change production from one month to the next, because we would need to lay off/hire new workers and change factories/etc.

**How can we phrase this problem as a linear programming problem?**
Specifically, what variables do we want to use? What should the constraints be on those variables? And what does the objective function look like?

Variables:

- $d_i$ - These are not variables we are maximizing over, they are inputs to the problem. This specifies the amount of ice cream demanded at month $i$ in tons.
- $x_i$ - the amount of production in month $i$ in tons.
- $s_i$ - the amount of surplus after month $i$ in tons.

Constraints:

- $x_i \geq 0$ - we can't have negative production in any month.
- $s_i \geq 0$ - there must be enough production to meet demand at each month.
- $s_{12} = 0$ - the amount of production in month $i$ in tons.
- $x_i + s_{i-1} - s_i = d_i$ - this asserts that the $s_i$ are updated correctly.
- $s_0 = 0$ - No previous storage existed. Could be modified if we know the history of the company
- $x_0 = 0$ - No production occurred at month 0. Could be modified if we know the history of the company

Objective function: We wish to maximize the profit or equivalently minimize the cost. To do this, we use the objective function:

$$\sum_i c_1 s_i + c_2 |x_i - x_{i-1}|$$

**How can we account for the absolute value in this expression?**
Notice that we do not yet have a linear program. This is because $|x|$ is not a linear function in $x$. However, we can make a linear function that will behave exactly like $|x|$ if we add

additional variables. To do this, we add two sets of variables, $y_i$ and $z_i$. Here $y_i$ and $z_i$ are defined so that $y_i$ is the increase in production from one month to the next, and $z_i$ is the decrease in production. In this light, we see that

$$|x_i - x_{i-1}| = y_i + z_i \text{ and } x_i - x_{i-1} = y_i - z_i$$

Then, our linear program becomes:

Variables:

- $d_i$ - These are not variables we are maximizing over, they are inputs to the problem. This specifies the amount of ice cream demanded at month $i$ in tons.
- $x_i$ - the amount of production in month $i$ in tons.
- $s_i$ - the amount of surplus after month $i$ in tons.
- $y_i$ - the increase in production from month $i-1$ to month $i$ in tons.
- $z_i$ - the decrease in production from month $i-1$ to month $i$ in tons.

Constraints:

- $x_i \geq 0$ - we can't have negative production in any month.
- $y_i \geq 0$ - we can't have negative increase.
- $z_i \geq 0$ - we can't have negative decrease.
- $s_i \geq 0$ - there must be enough production to meet demand at each month.
- $s_{12} = 0$ - the amount of production in month $i$ in tons.
- $s_0 = 0$ - No previous storage existed. Could be modified if we know the history of the company
- $x_0 = 0$ - No production occurred at month 0. Could be modified if we know the history of the company
- $x_i + s_{i-1} - s_i = d_i$ - this asserts that the $s_i$ are updated correctly.

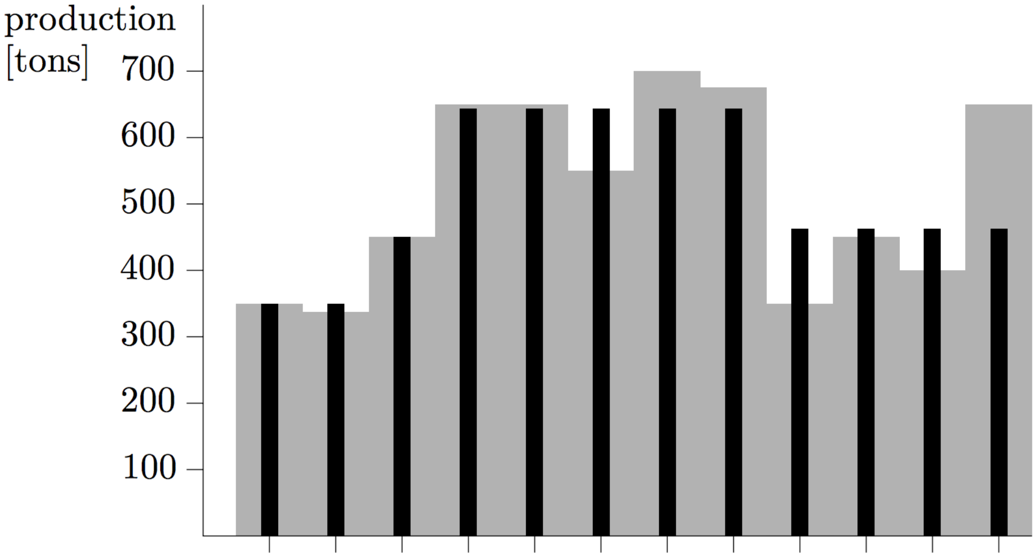


Figure 2.2: Solution to the ice cream company problem.

- $y_i \geq x_i - x_{i-1}$ - if there was an increase, this ensures that $y_i$ is at least as large as the increase
- $z_i \geq x_{i-1} - x_i$ - if there was a decrease, this ensures that $z_i$ is at least as large as the decrease

Objective function: We wish to maximize the profit or equivalently minimize the cost. To do this, we use the objective function:

$$\sum_i c_1 s_i + c_2(y_i + z_i)$$

The book actually provides a solution to this problem with $c_1 = 20$ and $c_2 = 50$, which is shown in the chart above.