Linear programming
COMS20010 (Algorithms II)

John Lapinskas, University of Bristol

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;
- Manufacturing;

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;
- Manufacturing;
- Power provision;

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;
- Manufacturing;
- Power provision;
- Approximation algorithms;

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;
- Manufacturing;
- Power provision;
- Approximation algorithms;
- Planning entire economies.

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;
- Manufacturing;
- Power provision;
- Approximation algorithms;
- Planning entire economies. **(VERY BAD IDEA!)**

What is Linear Programming?

Linear programming is the single most fundamental technique for solving optimisation problems. It's used in:

- Agriculture;
- Nutrition;
- Transport;
- Manufacturing;
- Power provision;
- Approximation algorithms;
- Planning entire economies. **(VERY BAD IDEA!)**

These two videos are a very basic overview of a deep and rich theory.

As an example problem: which Warhammer models should Games Workshop produce in order to make as much money as possible?

Example application: Warhammer

Let's consider a vastly simplified problem with just two models:

Example application: Warhammer

Let's consider a vastly simplified problem with just two models:

The noise marine...

Let's consider a vastly simplified problem with just two models:

The noise marine...

and the doomwheel.

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.
- Similarly, their metal plant can turn out 4kg of finished parts per day. One noise marine contains 60g of metal, and one doomwheel contains 10g, so...

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.
- Similarly, their metal plant can turn out 4kg of finished parts per day. One noise marine contains 60g of metal, and one doomwheel contains 10g, so...**they require** $60N + 10D \leq 4000$.

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.
- Similarly, their metal plant can turn out 4kg of finished parts per day. One noise marine contains 60g of metal, and one doomwheel contains 10g, so...**they require** $60N + 10D \leq 4000$.
- They believe they can sell up to 100 noise marines and 50 doomwheels per day, but no more, so...

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.
- Similarly, their metal plant can turn out 4kg of finished parts per day. One noise marine contains 60g of metal, and one doomwheel contains 10g, so...**they require** $60N + 10D \leq 4000$.
- They believe they can sell up to 100 noise marines and 50 doomwheels per day, but no more, so...**they require** $N \leq 100$ **and** $D \leq 50$.

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.
- Similarly, their metal plant can turn out 4kg of finished parts per day. One noise marine contains 60g of metal, and one doomwheel contains 10g, so...**they require** $60N + 10D \leq 4000$.
- They believe they can sell up to 100 noise marines and 50 doomwheels per day, but no more, so...**they require** $N \leq 100$ **and** $D \leq 50$.
- Games Workshop cannot produce a negative amount of miniatures, so...

Let N be the number of noise marines Games Workshop produces per day, and let D be the number of doomwheels. Suppose the numbers are as follows:

- Games Workshop makes a profit of £4 per noise marine and £10 per doomwheel, so...**they wish to maximise** $4N + 10D$.
- Their plastic plant can turn out 5kg of finished parts per day. One noise marine contains 5g of plastic, and one doomwheel contains 100g, so...**they require** $5N + 100D \leq 5000$.
- Similarly, their metal plant can turn out 4kg of finished parts per day. One noise marine contains 60g of metal, and one doomwheel contains 10g, so...**they require** $60N + 10D \leq 4000$.
- They believe they can sell up to 100 noise marines and 50 doomwheels per day, but no more, so...**they require** $N \leq 100$ **and** $D \leq 50$.
- Games Workshop cannot produce a negative amount of miniatures, so...**they require** $N,\ D \geq 0$.

More succinctly, the problem is:

$$\begin{aligned} 4N + 10D &\to \max, \text{ subject to} \\ 5N + 100D &\leq 5000; \\ 60N + 10D &\leq 4000; \\ N &\leq 100; \\ D &\leq 50; \\ N,\ D &\geq 0. \end{aligned}$$

More succinctly, the problem is:

$$
\begin{aligned}
4N + 10D &\to \max, \text{ subject to} \\
5N + 100D &\leq 5000; \\
60N + 10D &\leq 4000; \\
N &\leq 100; \\
D &\leq 50; \\
N,\ D &\geq 0.
\end{aligned}
$$

We can write this in matrix form:

$$
\begin{gathered}
4N + 10D \to \max, \text{ subject to} \\
\begin{pmatrix} 5 & 100 \\ 60 & 10 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} N \\ D \end{pmatrix} \leq \begin{pmatrix} 5000 \\ 4000 \\ 100 \\ 50 \end{pmatrix}; \\
N,\ D \geq 0.
\end{gathered}
$$

The formal definition

$$4N + 10D \to \max, \text{ subject to}$$

$$\begin{pmatrix} 5 & 100 \\ 60 & 10 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} N \\ D \end{pmatrix} \leq \begin{pmatrix} 5000 \\ 4000 \\ 100 \\ 50 \end{pmatrix};$$

$$N,\ D \geq 0.$$

The formal definition

$$4N + 10D \to \max, \text{ subject to}$$

$$\begin{pmatrix} 5 & 100 \\ 60 & 10 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} N \\ D \end{pmatrix} \leq \begin{pmatrix} 5000 \\ 4000 \\ 100 \\ 50 \end{pmatrix};$$

$$N,\ D \geq 0.$$

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

For example, $(2, 0, 1) \geq (0, 0, 0)$, but $(2, 0, 1) \not\geq (0, 1, 0)$.

The formal definition

$$4N + 10D \to \max, \text{ subject to}$$

$$\begin{pmatrix} 5 & 100 \\ 60 & 10 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} N \\ D \end{pmatrix} \le \begin{pmatrix} 5000 \\ 4000 \\ 100 \\ 50 \end{pmatrix};$$

$$N,\ D \ge 0.$$

Notation: We say $\vec{x} \le \vec{y}$ iff $\vec{x}_i \le \vec{y}_i$ for **all** i, and similarly for $\vec{x} \ge \vec{y}$.

For example, $(2,0,1) \ge (0,0,0)$, but $(2,0,1) \not\ge (0,1,0)$.
Despite this, we **also** have $(2,0,1) \not\le (0,1,0)$; they are incomparable.

The formal definition

$$4N + 10D \to \max, \text{ subject to}$$

$$\begin{pmatrix} 5 & 100 \\ 60 & 10 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} N \\ D \end{pmatrix} \leq \begin{pmatrix} 5000 \\ 4000 \\ 100 \\ 50 \end{pmatrix};$$

$$N, D \geq 0.$$

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

For example, $(2, 0, 1) \geq (0, 0, 0)$, but $(2, 0, 1) \not\geq (0, 1, 0)$.
Despite this, we **also** have $(2, 0, 1) \not\leq (0, 1, 0)$; they are incomparable.

Problem statement: We are given a linear **objective function** $f: \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

Is there always a solution?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f\colon \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

Is there always a solution?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f: \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

We say a $\vec{x} \in \mathbb{R}^n$ is a **feasible** solution to a linear program if $\vec{x} \geq \vec{0}$ and $A\vec{x} \leq \vec{b}$, and an **optimal** solution if $f(\vec{y}) \leq f(\vec{x})$ for all feasible $y \in \mathbb{R}^n$.

Is there always a solution?

Notation: We say $\vec{x} \le \vec{y}$ iff $\vec{x}_i \le \vec{y}_i$ for **all** i, and similarly for $\vec{x} \ge \vec{y}$.

Problem statement: We are given a linear objective function $f\colon \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \le \vec{b}$ and $\vec{x} \ge \vec{0}$.

We say a $\vec{x} \in \mathbb{R}^n$ is a **feasible** solution to a linear program if $\vec{x} \ge \vec{0}$ and $A\vec{x} \le \vec{b}$, and an **optimal** solution if $f(\vec{y}) \le f(\vec{x})$ for all feasible $y \in \mathbb{R}^n$.

Sometimes there is **no** optimal solution, for two reasons:

1. Sometimes the constraints are so tight they rule out any feasible solutions at all, e.g. $x \to \max$ subject to $x \le -1$ and $x \ge 0$.

Is there always a solution?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

We say a $\vec{x} \in \mathbb{R}^n$ is a **feasible** solution to a linear program if $\vec{x} \geq \vec{0}$ and $A\vec{x} \leq \vec{b}$, and an **optimal** solution if $f(\vec{y}) \leq f(\vec{x})$ for all feasible $y \in \mathbb{R}^n$.

Sometimes there is **no** optimal solution, for two reasons:

1. Sometimes the constraints are so tight they rule out any feasible solutions at all, e.g. $x \to \max$ subject to $x \leq -1$ and $x \geq 0$.
2. Sometimes the constraints are so loose that there are feasible solutions with $f(\vec{x})$ arbitrarily large, e.g. $x \to \max$ subject to $x \geq 0$. We call these problems **unbounded**.

Is there always a solution?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

We say a $\vec{x} \in \mathbb{R}^n$ is a **feasible** solution to a linear program if $\vec{x} \geq \vec{0}$ and $A\vec{x} \leq \vec{b}$, and an **optimal** solution if $f(\vec{y}) \leq f(\vec{x})$ for all feasible $y \in \mathbb{R}^n$.

Sometimes there is **no** optimal solution, for two reasons:

1. Sometimes the constraints are so tight they rule out any feasible solutions at all, e.g. $x \to \max$ subject to $x \leq -1$ and $x \geq 0$.
2. Sometimes the constraints are so loose that there are feasible solutions with $f(\vec{x})$ arbitrarily large, e.g. $x \to \max$ subject to $x \geq 0$. We call these problems **unbounded**.

But these are the only two things that can go wrong — any bounded linear program with at least one feasible solution has an optimal solution.

What about other "linear" problems?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

This statement seems quite restrictive. What about:

- Minimisation problems?

What about other "linear" problems?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

This statement seems quite restrictive. What about:

- Minimisation problems?
- $=$ or $\geq$ constraints?

What about other "linear" problems?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f\colon \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

This statement seems quite restrictive. What about:

- Minimisation problems?
- $=$ or $\geq$ constraints?
- Allowing the variables to be negative?

What about other “linear” problems?

Notation: We say $\vec{x} \leq \vec{y}$ iff $\vec{x}_i \leq \vec{y}_i$ for **all** i, and similarly for $\vec{x} \geq \vec{y}$.

Problem statement: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

This statement seems quite restrictive. What about:

- Minimisation problems?
- $=$ or $\geq$ constraints?
- Allowing the variables to be negative?

All of these can be implemented in the above framework, which is known as **standard form**.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\begin{aligned} 4x - 5y + z \to \min \;\; &\text{subject to} \\ x + y + z = 5&; \\ x + 2y \geq 2&; \\ x, z \geq 0&. \end{aligned}$$

Standard form: We are given a linear objective function $f\colon \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \le \vec{b}$ and $\vec{x} \ge \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\mathbf{4x - 5y + z \to min} \text{ subject to}$$
$$x + y + z = 5;$$
$$x + 2y \ge 2;$$
$$x, z \ge 0.$$

Minimisation problems: $f(\vec{x})$ is as small as possible if and only if $-f(\vec{x})$ is as large as possible.

So $4x - 5y + z \to \text{min}$ is equivalent to $-4x + 5y - z \to \text{max}$.

Standard form: We are given a linear objective function $f: \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\mathbf{-4x + 5y - z \to max} \text{ subject to}$$
$$x + y + z = 5;$$
$$x + 2y \geq 2;$$
$$x,\ z \geq 0.$$

Minimisation problems: $f(\vec{x})$ is as small as possible if and only if $-f(\vec{x})$ is as large as possible.

So $4x - 5y + z \to \text{min}$ is equivalent to $-4x + 5y - z \to \text{max}$.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\begin{aligned} -4x + 5y - z \to \max \;\; & \text{subject to} \\ \boldsymbol{x + y + z = 5}&; \\ x + 2y \geq 2&; \\ x, z \geq 0&. \end{aligned}$$

= constraints: $\sum_j a_{ij}x_j = b_i$ if and only if $\sum_j a_{ij}x_j \geq b_i$ **and** $\sum_i a_i x_i \leq b_i$.

So $x + y + z = 5$ is equivalent to $x + y + z \leq 5$ **and** $x + y + z \geq 5$.

Standard form: We are given a linear objective function $f\colon \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$
\begin{aligned}
-4x + 5y - z \to \max \ \text{subject to}& \\
\boldsymbol{x + y + z \leq 5}&; \\
\boldsymbol{x + y + z \geq 5}&; \\
x + 2y \geq 2&; \\
x,\, z \geq 0&.
\end{aligned}
$$

= constraints: $\sum_j a_{ij}x_j = b_i$ if and only if $\sum_j a_{ij}x_j \geq b_i$ **and** $\sum_i a_i x_i \leq b_i$.

So $x + y + z = 5$ is equivalent to $x + y + z \leq 5$ **and** $x + y + z \geq 5$.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\begin{aligned} -4x + 5y - z \to \max \text{ subject to} \\ x + y + z \leq 5; \\ \boldsymbol{x + y + z \geq 5}; \\ \boldsymbol{x + 2y \geq 2}; \\ x, z \geq 0. \end{aligned}$$

$\geq$ **constraints:** $\sum_j a_{ij}x_j \geq b_i$ if and only if $-\sum_j a_{ij}x_j \leq -b_i$.

So $x + 2y \geq 2$ is equivalent to $-x - 2y \leq -2$, and $x + y + z \geq 5$ is equivalent to $-x - y - z \leq -5$.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\begin{aligned} -4x + 5y - z \to \max \ \text{subject to} \\ x + y + z \leq 5; \\ \boldsymbol{-x - y - z \leq -5}; \\ \boldsymbol{-x - 2y \leq -2}; \\ x, z \geq 0. \end{aligned}$$

$\geq$ **constraints:** $\sum_j a_{ij}x_j \geq b_i$ if and only if $-\sum_j a_{ij}x_j \leq -b_i$.

So $x + 2y \geq 2$ is equivalent to $-x - 2y \leq -2$, and $x + y + z \geq 5$ is equivalent to $-x - y - z \leq -5$.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\begin{aligned} -4x + 5\mathbf{y} - z &\to \max \text{ subject to} \\ x + \mathbf{y} + z &\leq 5; \\ -x - \mathbf{y} - z &\leq -5; \\ -x - 2\mathbf{y} &\leq -2; \\ x,\, z &\geq 0. \end{aligned}$$

Removing non-negativity: If y doesn't have to be non-negative, we can replace it by $y_1 - y_2$ where $y_1, y_2 \geq 0$. We think of y_1 as the positive part and y_2 as the negative part.

There will be feasible solutions with both $y_1 > 0$ and $y_2 > 0$, but this doesn't matter — any optimal solution of the old problem will be an optimal solution of the new one and vice versa.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \leq \vec{b}$ and $\vec{x} \geq \vec{0}$.

As an example, let's turn the following LP into standard form:

$$\begin{aligned} -4x + 5(\mathbf{y_1} - \mathbf{y_2}) - z &\to \max \text{ subject to} \\ x + (\mathbf{y_1} - \mathbf{y_2}) + z &\leq 5; \\ -x - (\mathbf{y_1} - \mathbf{y_2}) - z &\leq -5; \\ -x - 2(\mathbf{y_1} - \mathbf{y_2}) &\leq -2; \\ x, \mathbf{y_1}, \mathbf{y_2}, z &\geq 0. \end{aligned}$$

Removing non-negativity: If y doesn't have to be non-negative, we can replace it by $y_1 - y_2$ where $y_1, y_2 \geq 0$. We think of y_1 as the positive part and y_2 as the negative part.

There will be feasible solutions with both $y_1 > 0$ and $y_2 > 0$, but this doesn't matter — any optimal solution of the old problem will be an optimal solution of the new one and vice versa.

Standard form: We are given a linear objective function $f : \mathbb{R}^n \to \mathbb{R}$, an $m \times n$ matrix A, and an m-dimensional vector $\vec{b} \in \mathbb{R}^m$. The desired output is a vector $\vec{x} \in \mathbb{R}^n$ maximising $f(\vec{x})$ subject to $A\vec{x} \le \vec{b}$ and $\vec{x} \ge \vec{0}$.

As an example, let's turn the following LP into standard form:

$$-4x + 5y_1 - 5y_2 - z \to \max \text{ subject to}$$

$$\begin{pmatrix} 1 & 1 & -1 & 1 \\ -1 & -1 & 1 & -1 \\ -1 & -2 & 2 & 0 \end{pmatrix} \begin{pmatrix} x \\ y_1 \\ y_2 \\ z \end{pmatrix} \le \begin{pmatrix} 5 \\ -5 \\ -2 \end{pmatrix};$$

$$x,\ y_1,\ y_2,\ x \ge 0.$$

The problem is now in standard form! And these techniques are fully general.

So we have **reduced** the problem of solving a general linear program, which might have a minimisation goal, $=$ or $\le$ constraints, and/or negative variables, to that of solving a linear program in standard form.

That makes it easier to find an algorithm!