Mathematical Methods of Algorithmics

Chapter 1: Introduction to Linear Programming

Phillip Keldenich Ahmad Moradi

Department of Computer Science
Algorithms Department
TU Braunschweig

November 7, 2023

Meet Your Teachers

Ahmad Moradi
I am a PostDoc researcher at TU Braunschweig.

Email: `moradi@ibr.cs.tu-bs.de`

Phillip Keldenich
I am a PostDoc researcher at TU Braunschweig.

Email: `keldenich@ibr.cs.tu-bs.de`

INTRODUCTION

MOTIVATION

DEFINITIONS

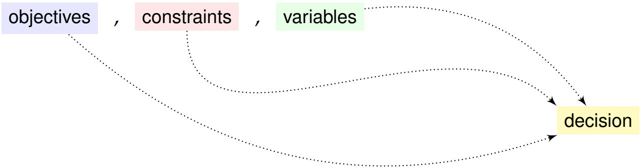
ORGANIZATION

- As usual, the module has a „Prüfungsleistung" and a „Studienleistung".
- The „Prüfungsleistung" will most likely be an oral exam, depending mostly on the number of participants. The „Prüfungsleistung" determines your grade.
- The „Studienleistung" is tied to the homework sheets. We will start homework sheets next week.
- You have two weeks to solve each homework assignment.
- As usual, there is a lecture (one per week) and a tutorial class (one per week, every other week being dedicated to homework discussion). The lecture is where the main content is presented. The tutorial adds additional content, practical stuff, shows applications, examples, and discusses questions related to the content.
- There is a mailing list and a course website. Please refer to that site instead of QIS/StudIP for information. Please sign up for the mailing list; you might miss important announcements otherwise.
  - `https://www.ibr.cs.tu-bs.de/courses/ws2324/mma/`
  - `https://lists.ibr.cs.tu-bs.de/postorius/lists/mma.ibr.cs.tu-bs.de`

CONTENT

What is this course about?

The mathematics behind making optimal decisions [1]

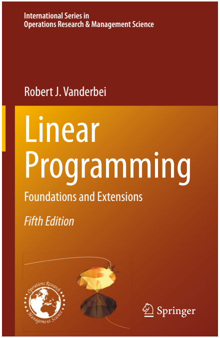


[1] https://stellato.io/downloads/teaching/orf522/01_lecture.pdf

LITERATURE

The main reference for this course:
[V] R. J. Vanderbei. Linear Programming: Foundations and Extensions. Springer Nature (2020).
Can be accessed through SpringerLink from the university network:

https://link.springer.com/book/10.1007/978-3-030-39415-8



MANAGING A PRODUCTION FACILITY

Consider a production facility which is capable of producing a variety of products, say n products. We enumerate these products as $1, 2, \ldots, n$.

These products are made from certain raw materials. Suppose that there are m different raw materials, which again we simply enumerate as $1, 2, \ldots, m$.

MANAGING A PRODUCTION FACILITY

Further properties:

- The facility has, for each raw material $i = 1, 2, \ldots, m$, a known amount, say b_i, on hand.
- Each raw material has, at this moment in time, a known unit market value. We denote the unit value of the ith raw material by ρ_i.
- Producing one unit of product j requires a certain known amount, say a_{ij} units, of raw material i.
- The jth final product can be sold at the known market price of σ_j dollars per unit.

MANAGING A PRODUCTION FACILITY

Let us assume that the production manager decides to produce one unit of the jth product.

- Revenue of one unit of product j is σ_j
- Cost of producing one unit of j is $\sum_{i=1}^{m} \rho_i a_{ij}$

Therefore, the net revenue associated with the production of one unit of j is the difference between the revenue and the cost.

$$c_j = \sigma_j - \sum_{i=1}^{m} \rho_i a_{ij}, \quad j = 1, 2, \ldots, n$$

For our optimization, we do not really care about the individual material costs; we only need to know the net revenue c_j associated with each product.

Managing a Production Facility

Let us capture the available information up to now:

$$
\begin{array}{c|ccc|c}
 & \text{product } 1 & \cdots & \text{product } n & \\
 & c_1 & \cdots & c_n & \\
\text{raw material } 1 & a_{11} & \cdots & a_{1n} & b_1 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\text{raw material } m & a_{m1} & \cdots & a_{mn} & b_m
\end{array}
$$

Maximizing Revenue

The problem we wish to consider is the one faced by the companies' production manager.

She asks:

> How to use the raw materials and get best possible net revenue?

Maximizing Revenue

The problem we wish to consider is the one faced by the companies' production manager.

She asks:

How to use the raw materials and get best possible net revenue?

Let x_j be the amount of the jth product she decides to produce.

MAXIMIZING REVENUE

The problem we wish to consider is the one faced by the companies' production manager.

She asks:

How to use the raw materials and get best possible net revenue?

Let x_j be the amount of the jth product she decides to produce.

$\rightarrow$ The net revenue corresponding to the production of x_j units of product j is simply $c_j x_j$.

Maximizing Revenue

The problem we wish to consider is the one faced by the companies' production manager.

She asks:

How to use the raw materials and get best possible net revenue?

Let x_j be the amount of the jth product she decides to produce.

- → The net revenue corresponding to the production of x_j units of product j is simply $c_j x_j$.
- → The total net revenue is

$$\sum_{j=1}^{n} c_j x_j .$$

MAXIMIZING REVENUE

The problem we wish to consider is the one faced by the companies' production manager.

She asks:

How to use the raw materials and get best possible net revenue?

Let x_j be the amount of the jth product she decides to produce.

→ The net revenue corresponding to the production of x_j units of product j is simply $c_j x_j$.

→ The total net revenue is

$$\sum_{j=1}^{n} c_j x_j .$$

→ Her goal is to find values x_j to maximize this quantity.

CONSTRAINTS

However, there are constraints on the production levels that she can assign.

CONSTRAINTS

However, there are constraints on the production levels that she can assign.

→ Each production quantity x_j must be nonnegative, so she has the constraints

$$x_j \geq 0, \quad j = 1, 2, \ldots, n.$$

CONSTRAINTS

However, there are constraints on the production levels that she can assign.

→ Each production quantity x_j must be nonnegative, so she has the constraints

$$x_j \geq 0, \quad j = 1, 2, \ldots, n.$$

→ She cannot produce more product than she has raw material for. The amount of raw material i consumed by a given production schedule is

$$\sum_{j=1}^{n} a_{ij} x_j,$$

so she must adhere to the following constraints

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, 2, \ldots, m.$$

Objective

$$\max \quad \sum_{j=1}^{n} c_j x_j$$

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \quad , \quad i = 1, 2, \cdots, m$$

$$x_j \geq 0 \quad , \quad j = 1, 2, \cdots, n$$

$$x_j \quad , \quad j = 1, 2, \cdots, n$$

Constraints

variables

EXAMPLE

Resource allocation in a toy factory. [2]

	toy 1	toy 2	toy 3	toy 4	toy 5	
	\$15	\$30	\$20	\$25	\$25	
1. Red paint	0	1	0	1	3	625
2. Blue paint	3	1	0	1	0	640
3. White paint	2	1	2	0	2	1100
4. Plastic	1	5	2	2	1	875
5. Wood	3	0	3	5	5	2200
6. Glue	1	2	3	2	3	1500

[2] https://www.exceldemy.com/allocating-resources-in-excel-using-solver/

EXAMPLE

	toy 1	toy 2	toy 3	toy 4	toy 5	
	\$15	\$30	\$20	\$25	\$25	
1. Red paint	0	1	0	1	3	625
2. Blue paint	3	1	0	1	0	640
3. White paint	2	1	2	0	2	1100
4. Plastic	1	5	2	2	1	875
5. Wood	3	0	3	5	5	2200
6. Glue	1	2	3	2	3	1500

$$\begin{aligned}
\max_{x} \quad & 15x_1 + 30x_2 + 20x_3 + 25x_4 + 25x_5 \\
s.t. \quad & 0x_1 + 1x_2 + 0x_3 + 1x_4 + 3x_5 \leq 625 \\
& 3x_1 + 1x_2 + 0x_3 + 1x_4 + 0x_5 \leq 640 \\
& 2x_1 + 1x_2 + 2x_3 + 0x_4 + 2x_5 \leq 1100 \\
& 1x_1 + 5x_2 + 2x_3 + 2x_4 + 1x_5 \leq 875 \\
& 3x_1 + 0x_2 + 3x_3 + 5x_4 + 5x_5 \leq 2200 \\
& 1x_1 + 2x_2 + 3x_3 + 2x_4 + 3x_5 \leq 1500 \\
& x_1, x_2, x_3, x_4, x_5 \geq 0
\end{aligned}$$

Blue paint cons.

Wood cons.

(Linear Programming formulation of the problem)

LINEAR PROGRAM

Let us capture important points observed up to now:

→ In the examples, there have been variables whose values are to be decided in some optimal fashion. These variables are referred to as *decision variables*. They are usually denoted as

$$x_j \quad , \quad j = 1, 2, \cdots, n$$

Linear Program

Let us capture important points observed up to now:

→ In the examples, there have been variables whose values are to be decided in some optimal fashion. These variables are referred to as *decision variables*. They are usually denoted as

$$x_j \ , \quad j = 1, 2, \cdots, n$$

→ The objective is always to *maximize or to minimize some linear function* of these decision variables

$$\zeta = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n.$$

This function is called the *objective function*.

LINEAR PROGRAM

Let us capture important points observed up to now:

- → In the examples, there have been variables whose values are to be decided in some optimal fashion. These variables are referred to as *decision variables*. They are usually denoted as

$$x_j \quad , \quad j = 1, 2, \cdots, n$$

- → The objective is always to *maximize or to minimize some linear function* of these decision variables

$$\zeta = c_1x_1 + c_2x_2 + \cdots + c_nx_n.$$

  This function is called the *objective function*.

- → In addition to the objective function, the examples also had constraints. The constraints consisted of either an equality or an inequality associated with some *linear combination* of the decision variables:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \left\{ \begin{array}{c} \leq \\ = \\ \geq \end{array} \right\} b$$

LINEAR PROGRAM

Let us capture important points observed up to now:

→ In the examples, there have been variables whose values are to be decided in some optimal fashion. These variables are referred to as *decision variables*. They are usually denoted as

$$x_j \quad , \quad j = 1, 2, \cdots, n$$

→ The objective is always to *maximize or to minimize some linear function* of these decision variables

$$\zeta = c_1x_1 + c_2x_2 + \cdots + c_nx_n.$$

This function is called the *objective function*.

→ In addition to the objective function, the examples also had constraints. The constraints consisted of either an equality or an inequality associated with some *linear combination* of the decision variables:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \left\{ \begin{array}{c} \leq \\ = \\ \geq \end{array} \right\} b$$

Note: No multiplication of decision variables with each other!

Converting between constraint types and objective function directions is straightforward!

→ An objective function in $\max$ sense could be converted to $\min$ and vice versa:

Converting between constraint types and objective function directions is straightforward!

→ An objective function in $\max$ sense could be converted to $\min$ and vice versa:

$$\max_{x}\ \zeta(x) = -\min_{x}\ -\zeta(x).$$

Converting between constraint types and objective function directions is straightforward!

→ An objective function in max sense could be converted to min and vice versa:

$$\max_x \zeta(x) = -\min_x -\zeta(x).$$

→ An inequality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$

can be converted to an equality constraint by adding a *nonnegative* variable, w, called *slack variable*:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + w = b, \quad w \geq 0.$$

Converting between constraint types and objective function directions is straightforward!

→ An objective function in $\max$ sense could be converted to $\min$ and vice versa:

$$\max_x \ \zeta(x) = -\min_x \ -\zeta(x).$$

→ An inequality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$

can be converted to an equality constraint by adding a *nonnegative* variable, w, called *slack variable*:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + w = b, \quad w \geq 0.$$

→ An equality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = b$$

can be converted to inequality form by

Converting between constraint types and objective function directions is straightforward!

→ An objective function in $\max$ sense could be converted to $\min$ and vice versa:

$$\max_x \ \zeta(x) = -\min_x \ -\zeta(x).$$

→ An inequality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$

can be converted to an equality constraint by adding a *nonnegative* variable, w, called *slack variable*:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + w = b, \quad w \geq 0.$$

→ An equality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = b$$

can be converted to inequality form by introducing two inequality constraints:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$
$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \geq b$$

Converting between constraint types and objective function directions is straightforward!

→ An objective function in max sense could be converted to min and vice versa:

$$\max_x \zeta(x) = -\min_x -\zeta(x).$$

→ An inequality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$

can be converted to an equality constraint by adding a *nonnegative* variable, w, called *slack variable*:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + w = b, \quad w \geq 0.$$

→ An equality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = b$$

can be converted to inequality form by introducing two inequality constraints:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$
$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \geq b$$

- A $\geq$-constraint can be transformed to $\leq$ by

Converting between constraint types and objective function directions is straightforward!

→ An objective function in max sense could be converted to min and vice versa:

$$\max_x \zeta(x) = -\min_x -\zeta(x).$$

→ An inequality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$

can be converted to an equality constraint by adding a *nonnegative* variable, w, called *slack variable*:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + w = b, \quad w \geq 0.$$

→ An equality constraint

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = b$$

can be converted to inequality form by introducing two inequality constraints:

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b$$
$$a_1x_1 + a_2x_2 + \cdots + a_nx_n \geq b$$

- A $\geq$-constraint can be transformed to $\leq$ by negating both sides:

$$\sum_i a_ix_i \geq b_i \Leftrightarrow \sum_i -a_ix_i \leq -b_i.$$

STANDARD FORM

There is no a priori preference for how one poses the constraints (as long as they are linear, of course). However, from a mathematical point of view, there is a preferred presentation.

Linear program in *Standard Form* representation:

- Consider a $\max$ problem,
- pose the inequalities in $\leq$-form,
- stipulate that all the decision variables be nonnegative.

$$
\begin{aligned}
\max_{x} \quad & c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \\
\text{subject to} \quad & a_{11} x_1 + a_{12} x_2 + \cdots + a_{1n} x_n \leq b_1 \\
& a_{21} x_1 + a_{22} x_2 + \cdots + a_{2n} x_n \leq b_2 \\
& \qquad\qquad \vdots \\
& a_{m1} x_1 + a_{m2} x_2 + \cdots + a_{mn} x_n \leq b_m \\
& x_1, x_2, \ldots, x_n \geq 0.
\end{aligned}
$$

SOLUTIONS & FEASIBILITY

A proposal of *specific values* for the decision variables is called a *solution*.

- A solution $(x_1, x_2, \cdots, x_n)$ is called *feasible* if it satisfies all of the constraints.
- It is called *optimal* if, in addition to feasibility, it attains the desired maximum.

Solutions & Feasibility

A proposal of *specific values* for the decision variables is called a *solution*.

- A solution $(x_1, x_2, \cdots, x_n)$ is called *feasible* if it satisfies all of the constraints.
- It is called *optimal* if, in addition to feasibility, it attains the desired maximum.

Some problems are just simply infeasible. Consider

$$\begin{aligned} \max_{x} \quad & 5x_1 + 4x_2 \\ s.t. \quad & x_1 + x_2 \leq 2 \\ & -2x_1 - 2x_2 \leq -9 \\ & x_1, x_2 \geq 0. \end{aligned}$$

SOLUTIONS & FEASIBILITY

A proposal of *specific values* for the decision variables is called a *solution*.

- A solution $(x_1, x_2, \cdots, x_n)$ is called *feasible* if it satisfies all of the constraints.
- It is called *optimal* if, in addition to feasibility, it attains the desired maximum.

Some problems are just simply infeasible. Consider

$$\begin{aligned} \max_x \quad & 5x_1 + 4x_2 \\ s.t. \quad & x_1 + x_2 \leq 2 \\ & -2x_1 - 2x_2 \leq -9 \\ & x_1, x_2 \geq 0. \end{aligned}$$

The second constraint implies that $x_1 + x_2 \geq 4.5$, which contradicts the first constraint.

SOLUTIONS & FEASIBILITY

A proposal of *specific values* for the decision variables is called a *solution*.

- A solution $(x_1, x_2, \cdots, x_n)$ is called *feasible* if it satisfies all of the constraints.
- It is called *optimal* if, in addition to feasibility, it attains the desired maximum.

Some problems are just simply infeasible. Consider

$$\begin{aligned} \max_{x} \quad & 5x_1 + 4x_2 \\ s.t. \quad & x_1 + x_2 \leq 2 \\ & -2x_1 - 2x_2 \leq -9 \\ & x_1, x_2 \geq 0. \end{aligned}$$

The second constraint implies that $x_1 + x_2 \geq 4.5$, which contradicts the first constraint.

- If a problem has no feasible solution, then the problem itself is called *infeasible*.

UNBOUNDEDNESS

At the other extreme from infeasible problems, one finds unbounded problems.

- A problem is *unbounded* if it has feasible solutions with arbitrarily large objective values.

UNBOUNDEDNESS

At the other extreme from infeasible problems, one finds unbounded problems.

– A problem is *unbounded* if it has feasible solutions with arbitrarily large objective values.

$$\begin{aligned} \max_{x} \quad & x_1 - 4x_2 \\ s.t. \quad & -2x_1 + x_2 \leq -1 \\ & -x_1 - 2x_2 \leq -2 \\ & x_1, x_2 \geq 0 \end{aligned}$$

UNBOUNDEDNESS

At the other extreme from infeasible problems, one finds unbounded problems.

– A problem is *unbounded* if it has feasible solutions with arbitrarily large objective values.

$$\begin{aligned} \max_x \quad & x_1 - 4x_2 \\ s.t. \quad & -2x_1 + x_2 \leq -1 \\ & -x_1 - 2x_2 \leq -2 \\ & x_1, x_2 \geq 0 \end{aligned}$$

$\rightsquigarrow$ Set x_2 to zero and let x_1 be arbitrarily large.

$\rightsquigarrow$ As long as $x_1 \geq 2$, the solution will be feasible, and

$\rightsquigarrow$ As x_1 gets large the objective function does too.

UNBOUNDEDNESS

At the other extreme from infeasible problems, one finds unbounded problems.

– A problem is *unbounded* if it has feasible solutions with arbitrarily large objective values.

$$\begin{aligned}\max_x \quad & x_1 - 4x_2 \\ s.t. \quad & -2x_1 + x_2 \leq -1 \\ & -x_1 - 2x_2 \leq -2 \\ & x_1, x_2 \geq 0\end{aligned}$$

$\rightsquigarrow$ Set x_2 to zero and let x_1 be arbitrarily large.

$\rightsquigarrow$ As long as $x_1 \geq 2$, the solution will be feasible, and

$\rightsquigarrow$ As x_1 gets large the objective function does too.

In addition to finding optimal solutions to linear programming problems, we are going to *detect* when a problem is infeasible or unbounded.

Existence of Optimal Solutions

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Existence of Optimal Solutions

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

EXISTENCE OF OPTIMAL SOLUTIONS

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

Let X be the set of feasible solutions.

EXISTENCE OF OPTIMAL SOLUTIONS

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

Let X be the set of feasible solutions.

$\rightarrow$ The objective function c is bounded on X, i.e., has a supremum s (for $\max$).

Existence of Optimal Solutions

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

Let X be the set of feasible solutions.

$\rightarrow$ The objective function c is bounded on X, i.e., has a supremum s (for $\max$).

$\rightarrow$ The objective function c is linear, so it maps the closed set X to a closed set $c[X]$.

EXISTENCE OF OPTIMAL SOLUTIONS

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

Let X be the set of feasible solutions.

$\rightarrow$ The objective function c is bounded on X, i.e., has a supremum s (for $\max$).

$\rightarrow$ The objective function c is linear, so it maps the closed set X to a closed set $c[X]$.

$\rightarrow$ By the definition of $\sup$, s is in $c[X]$ or an accumulation point of $c[X]$. Since $c[X]$ is closed, $s \in c[X]$ in either case.

EXISTENCE OF OPTIMAL SOLUTIONS

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

Let X be the set of feasible solutions.

→ The objective function c is bounded on X, i.e., has a supremum s (for $\max$).

→ The objective function c is linear, so it maps the closed set X to a closed set $c[X]$.

→ By the definition of $\sup$, s is in $c[X]$ or an accumulation point of $c[X]$. Since $c[X]$ is closed, $s \in c[X]$ in either case.

Why do we not allow $<$ and $>$ constraints in linear programs?

EXISTENCE OF OPTIMAL SOLUTIONS

So far, we know linear programs can be infeasible or unbounded. Do they always have an optimal solution if neither is true?

Answer: Yes!

Why?

Let X be the set of feasible solutions.

→ The objective function c is bounded on X, i.e., has a supremum s (for $\max$).

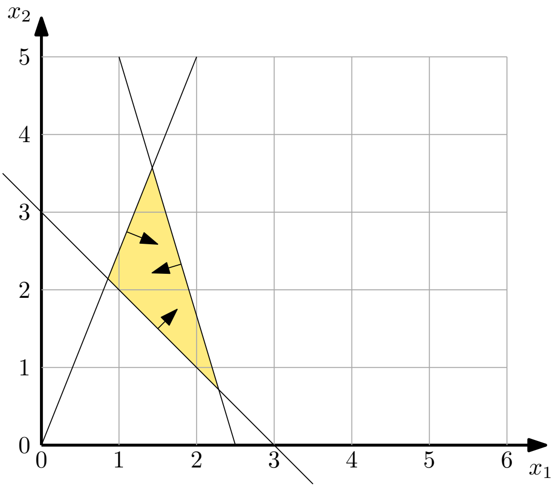
→ The objective function c is linear, so it maps the closed set X to a closed set $c[X]$.

→ By the definition of $\sup$, s is in $c[X]$ or an accumulation point of $c[X]$. Since $c[X]$ is closed, $s \in c[X]$ in either case.

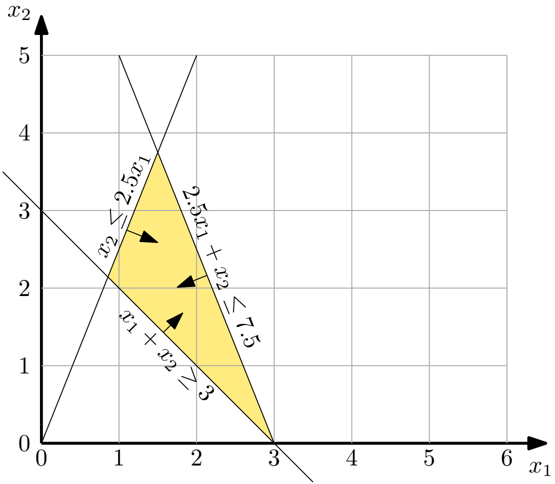
Why do we not allow $<$ and $>$ constraints in linear programs?
Because X would not be closed: $\max x$ s.t. $x < 1$?

GEOMETRY

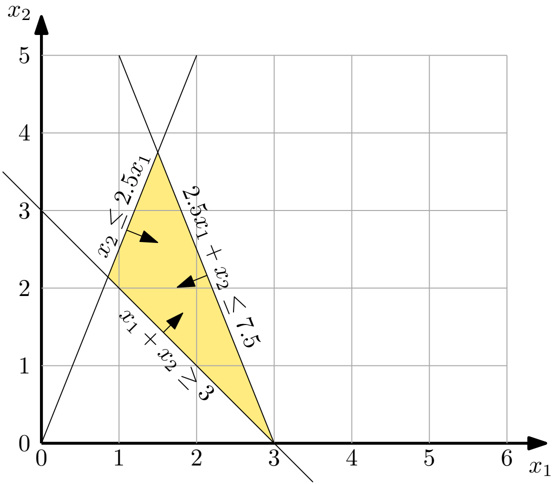
Find linear inequalities whose intersection makes the yellow region (feasible space).



Geometry



GEOMETRY



Up next: An algorithm to solve linear programs!