**Coursera – Discrete Optimization**

**LP 1: Intuition, Convexity, Geometric View (**[**link**](https://www.coursera.org/learn/discrete-optimization/lecture/UU9w9/lp-1-intuition-convexity-geometric-view)**)**

This lecture covers three topics:

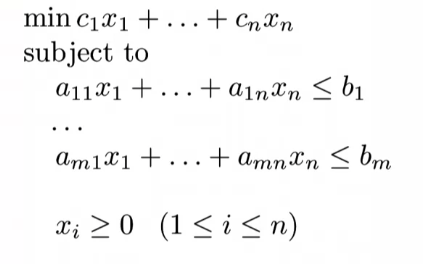
* What is Linear Programming (LP)?
* Convexity and its importance in LP
* Geometric representation of LP problems

*What is Linear Programming?*

Invented by George Dantzig in 1947, LP is one of the fundamental tools in *combinatorial optimization* (the topic of finding the optimal solution from a set of feasible solutions which is too large to exhaustively search via brute force).

A nice characteristic of LP is that all solutions derived from LP can be represented both geometrically and algebraically. Typically, when understanding an LP solution, we’ll go back and forth between the algebraic and geometric representations of the problem. Geometric representations allow us to visualize the problem, while algebraic representations allows us to distill the problem into an equation.

A linear program can be generalized as such:

The top equation represents the linear *objective function* which needs to be minimized/optimized. This objective function is subject to a set of linear inequalities called *constraints*. A few characteristics of the equations above:

* All variables need to be non-negative ()
* All of the in the equation above are all variables represent the real numbers included in the problem/dataset
* All of , , and are constants of the algebraic equations included in the problem (note that and are coefficients)
* Using the representation above, there are variables and constraints

The characteristics above naturally lead to these questions about LP:

* Can I maximize the objective function?

Yes. To do this, we can negate the objective function, and minimize this new equation –

* What should I do if a variable can take negative values?

You can replace with two non-negative values, whose difference can be negative. Replacing with *everywhere* in your linear program can enable negative values for your variables.

* Instead of an inequality constraint, what if I have an equality constraint?

This is an easy fix; we can just provide two inequality constraints – one which is the constant, another which is the constant, resulting in an equality constraint.

* What if my variable only takes integer values?

This can’t be handled directly by LP. This is another domain of optimization called *mixed integer linear programming* which is covered in other parts of the course.

* What if I have a non-linear constraint?

This also is not handled in LP. It is called *linear* programming after all ☺