

What is this course *really* about?

Linear program is a kind of constrained optimization problem. A constrained optimization problem asks you to maximize or minimize a (objective) function of several variables when the variables have to satisfy certain (in)equalities (constraints). You have encountered constrained optimization problems in Multivariable Calculus when you studied the method of Lagrange Multipliers. Problems in linear optimization are both simpler and more complicated than such problems you have seen before. They are simpler because the objective function and the constraints all have to be linear. But they are complicated because they have lots of variables and many times have other constraints (like integrality) which make them computationally difficult. The assumption about linearity makes Linear Algebra (instead of calculus) the primary tool to tackle such problems. In fact, a general LP asks for a vector x that minimizes the dot product $\mathbf{c} \cdot \mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$.

Each inequality specifies points on one side of a hyperplane (a half-space) in $\mathbb{R}^n$, where n is the number of variables. The intersection of all these half-spaces gives a convex, n -dimensional polytope (the feasible set). Linearity of LP guarantees that the optimal solution occurs at a corner (vertex) of the polytope. So in principle, to solve an LP, all we have to do is find all vertices of this polytope and evaluate the objective function on each of these points and pick the smallest.

However, this is not a practical algorithm because the number of vertices can be large. For example, the following n constraints in n variables: $0 \leq x_i \leq 1$, give $[0, 1]^n$, the n -dimensional hypercube as the polytope. And this hypercube has 2^n vertices. So it would take exponential number of steps to evaluate the objective function at all these possible solutions.

The simplex algorithm avoids this hassle by using the fact that if the objective function is not minimized at a particular vertex of the polytope then one of the neighboring vertices of the polytope will have a smaller objective function value. So, starting at a vertex we can take these “local” steps improving our objective function value and ultimately reach the optimal solution.

This is just the start of the story. Next obvious questions are: How do we pick a vertex to start our algorithm? How do we choose one of the neighboring vertices for the next step? Can we guarantee this will not take exponential number of steps? Can we easily check whether a particular solution is optimal or not? How can we solve very large LPs efficiently? How does the solution of the LP change if we change the objective function or the constraints by a small “amount”? How can we solve LPs that arise from Networks (like the Transportation problem)? How can we solve LPs that have the additional constraint that some of the variables must take integral value (Integer Programming)?

In this course, we will build a mathematical foundation through an interplay of geometry and linear algebra, that will allow us to understand all the underlying concepts of LPs stated above, and then explore the followup questions mentioned above. This will prepare you to apply these mathematical tools in non-trivial applications - LPs and their algorithms are used in building general-purpose optimization models, and general-purpose optimization algorithms. For example, Linear optimization techniques, including the simplex method, are widely used in real-life applications for decision-making in logistics and supply chains for businesses in transportation and distribution industries (like airlines, retailers, grocery chains, etc.). Problems involving tens of thousand of variables and millions of constraints are solved regularly. As discussed below, such non-trivial applications of mathematics (like Linear Optimization techniques) requires a deep understanding of the underlying mathematical structures and concepts.

According to Underwood Dudley, there are at least eight levels of mathematical understanding:

1. Being able to do arithmetic
2. Being able to substitute numbers in 'formulas'/ being able to state or use elementary properties of concepts
3. Given 'formulas'/ elementary properties of a concept, being able to get other 'formulas'/ elementary properties
4. Being able to understand hypotheses and conclusions of theorems
5. Being able to understand the proofs of theorems, step by step
6. Being able to *really* understand proofs of theorems: that is, seeing why the proof is as it is, and comprehending the underlying ideas of the proof and its relation to other proofs and theorems
7. Being able to generalize and extend theorems
8. Being able to see new relationships, and discover and prove entirely new theorems.

The word 'theorem' is used above in a very general sense - it can also represent algorithms and methods with a mathematical basis.

Levels 5 and 6 would be considered basic mathematical ability for Math majors. Non-trivial applications of Mathematics would lie in-between levels 6 and 7. While levels 7 and 8 constitute research in Mathematics. A lot of computer science, engineering, and physics is deep applied mathematics and requires understanding at or beyond levels 6 and 7.

Calculus courses focus on a mixture of 1 and 2. Math 230 (Introduction to Discrete Mathematics) focuses on 3 and 4. Math 332 (Elementary Linear Algebra) focuses on 3 and 4 with a bit of 5. In this course (Math 435/ 535), the focus is more on the upper part of levels 3, 4, 5, and 6, and the corresponding algorithms that arise out of this mathematical understanding. The aim is give you a firm foundation in levels up to 6, so that you can go onto levels 7 and 8, both as mathematicians and engineers (through non-trivial applications of Linear Optimization techniques).

I hope this course will help you make progress through these levels of mathematical understanding, and mathematical maturity. I would consider this a successful course, if you gain confidence in your ability to read, understand, and write mathematical arguments (including proofs), especially as compared to the beginning of the semester. And, you feel that you can read, understand, and apply any other topic/ technique in Linear Optimization that you might need later on in your career.

with best wishes,
Hemanshu Kaul