Lecture 1 – Linear Programs and Common Notation

Module 4 – Linear Programming Basics
CS/ISyE/ECE 524

Linear Programming Basics Module Learning Outcomes

By the end of this module, you should be able to…

- Give a formal definition of a linear program (LP)
- Transform any LP into standard form
- Graphically deduce whether an LP
  - has exactly one optimal solution
  - has infinitely many optimal solutions,
  - is infeasible, or
  - is unbounded.
- Begin to gain an intuition for the properties of LPs

Expressing Linear Programs in Math Notation

What *is* a linear program, anyway?

Variables

- Could be unrestricted:

  free for some

- Could have **box constraints**:

  for some

Objective function

Either:

Or:

Constraints

- Inequalities:
- Equalities:
- A combination of both

A **linear program** is an optimization model with:

- Real-valued variables ()
- A linear (or affine) objective function that we either maximize or minimize
- Affine constraints

Expressing a linear program in **standard form**

One of the most common ways to write down the variables, objective function, and constraints of a linear program is using **standard form:**

A linear program in **standard form** has the following properties:

- The *objective* is maximized
- All *constraints* are “less than or equal to”
- All *variables* are bounded below by 0 and are unbounded above

Note: There are many equivalent ways of writing down a linear program. We will only focus on a couple possibilities.

Standard Form Example

Recall the “Top Brass” model from the first module:

Rewrite as operations on matrices

Top Brass is already in standard form!

What to do if your LP isn’t in standard form?
(7 helpful tricks)

- Convert between max and min by taking the negative of the objective:
- Reverse direction of inequalities by flipping the sign:
- Turn equality into inequality by splitting it into two new inequalities:
- Turn inequalities into equalities by adding a **slack variable**:

What to do if your LP isn’t in standard form?
(7 helpful tricks continued)

- Replace free (unrestricted) variables with two new (non-negative) variables:
- Put variable bounds in constraints to create “free” variables:
- Shift bounds on bounded variables to make the lower bound:

Practice with Standard Form

Turn the following LP into standard form:

Question: What are the issues with this LP?

In Julia:
Standard Form

Note: Often some additional work is required after solving standard-form problems. In this example we have:

- original cost= -(new cost)
-

Standard Form Example in Julia

Posted on Canvas (link in “Linear Programming [basics]” Module)