Discrete Math And Algorithms Review

Muyao Xiao

Nov. 2024

Abstract

Discrete Math is a topic containing too many things, I would say it contains basic logic, number theory, probability theory, linear programming, set theory, graph theory, etc.

Considering the comprehensive nature of discrete math, I will pick topics that I don't have deep understanding or those I hope to review.

This note will basically be a speed run for now, but I expect to add more contents in the future if I have time to dive deep into certain topics.

Some contents will heavily depend on MIT 18.310 and book K. H. Rosen and K. Krithivasan, Discrete mathematics and its applications, 7. ed., Global ed. New York, NY: McGraw-Hill, 2013.

Contents

Chapter 1

Logic and Proofs

1.1 First-order logic

First-order logic – also called **predicate logic**, **predicate calculus**, **quantificational logic** (wikipedia).

Definition 1.1.1 (Proposition). A **proposition** is a declarative sentence (that is, a sentence that declares a fact) that is either true or false, but not both.

Example. Examples of proposition:

- Washington, D.C., is the capital of the United States of America.
- Toronto is the capital of Canada.
- $1 + 1 = 2$.
- $2 + 2 = 3$.

Propositional logic cannot adequately express the meaning of all statements in mathematics and in natural language.

Example. Suppose we know:
"Every computer connected to the university network is functioning properly".
No rules of propositional logic allow us to conclude the truth of the statement:
"MATH3 is functioning properly".
where MATH3 is one of the computers connected to the university network.

Chapter 2

Number Theory and Cryptography

Chapter 3

Counting

Chapter 4

Discrete Probability

Chapter 5

Graphs

Chapter 6

Linear Programming

Note. The content of this chapter is based on MIT 6.046J Design And Analysis Of Algorithms. Here is the video and note.

6.1 Politics Problem

Example. The example is about how to use the minimum money to win an election (win all the demographics).

The table is about 4 issues and 3 demographics, the number means how many voters can be obtained (bought) per dollar spent advertising the support of an issue:

Policy	Demographic		
	Urban	Suburban	Rural
Building roads	-2	5	3
Gun Control	8	2	-5
Farm Subsidies	0	0	10
Gasoline Tax	10	0	2
Population	100,000	200,000	50,000

Table 6.1: Votes per dollar spent on advertising, and population

6.1.1 Representation of the problem

Let x_1, x_2, x_3, x_4 denote the spent on each of the issues.

Rephrase our problem:

Minimize:

$$x_1 + x_2 + x_3 + x_4$$

Subject to:

$$-2x_1 + 8x_2 + 0x_3 + 10x_4 \geq 50,000 \quad \text{(Urban Majority (1))}$$
$$5x_1 + 2x_2 + 0x_3 + 0x_4 \geq 100,000 \quad \text{(Suburban Majority (2))}$$
$$3x_1 - 5x_2 + 10x_3 + 2x_4 \geq 25,000 \quad \text{(Rural Majority (3))}$$
$$x_1, x_2, x_3, x_4 \geq 0 \quad \text{(Can't unadvertise)}$$

The optimal solution is:

$$x_1 = \frac{2050000}{111} \tag{6.1}$$
$$x_2 = \frac{425000}{111} \tag{6.2}$$
$$x_3 = 0 \tag{6.3}$$
$$x_4 = \frac{625000}{111} \tag{6.4}$$
$$x_1 + x_2 + x_3 + x_4 = \frac{3100000}{111} \tag{6.5}$$

Certificate to this optimal solution:
Multiply the constraints inequalities with some magic numbers, and we have:

$$\frac{25}{222}(1) + \frac{46}{222}(2) + \frac{14}{222}(3) = x_1 + x_2 + \frac{140}{222}x_3 + x_4 \geq \frac{3100000}{111}$$

Considering that $x_1+x_2+x_3+x_4 \geq x_1+x_2+\frac{140}{222}x_3+x_4$, we know the number is already minimum. But how can we come to the solution and the verifying constants?

6.2 Standard Form of LP

Definition 6.2.1 (Linear Programming)**.** Minimize or maximum linear objective function subject to linear inequalities (or equations).
Variables:

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{bmatrix}$$

Objective functions: $\vec{c} \cdot \vec{x}$
Inequalities: $A\vec{x} \leq \vec{b}$
We want to maximize $\vec{c}\vec{x}$ and $\vec{x} \geq 0$

Remark. Any LP can be transformed into the standard form.

Definition 6.2.2 (Duality)**.** In mathematical optimization theory, duality or the duality principle is the principle that optimization problems may be viewed from either of two perspectives, the primal problem or the dual problem. If the primal is a minimization problem then the dual is a maximization problem (and vice versa). Wikipedia: Duality

Definition 6.2.3 (Duality Form)**.** There are 2 equivalent problems:
Primal Form:

- Maximize $\vec{c} \cdot \vec{x}$
- Subject to $A\vec{x} \leq \vec{b}, \vec{x} \geq 0$

Dual Form:

- Minimize $\vec{b} \cdot \vec{y}$
- Subject to $A^T\vec{y} \geq \vec{c}, \vec{y} \geq 0$

Remark. Notice that the exchange of $\vec{b}$ and $\vec{c}$. Also notice that both forms have their variables greater or equal to 0.

6.2.1 Transformation To Standard Form

Example. Minimize $-2x_1 + 3x_2$

Remark. The standard form is to maximize.

Transformation. Negate to $2x_1 - 3x_2$ and maximize ■

Example. Suppose x_j does NOT have a non-negativity constraint.

Remark. The standard form each x should be greater or equal to 0.

Transformation. Replace x_j with $x_j^{'} - x_j^{''}$ and $x_j^{'}, x_j^{''} \geq 0$ ■

Example. Equality constraint $x_1 + x_2 = 7$

Remark. We need less or equal to.

Transformation. $x_1 + x_2 \leq 7, -x_1 - x_2 \leq -7$ ■

By using the above techniques, we can always transform our LP problems into the standard form. Then we can use LP solvers to solve the problem.

6.3 Simplex Algorithm

The simplex algorithm works well in practice, but run in exponential time in the worst case. At high level, the algorithm works as **Gaussian elimination** on the inequalities or constraints.

Definition 6.3.1 (Simplex Algorithm)**.** The flow of simplex algorithm:

- Represent LP in "slack" form
- Convert one slack form into an equivalent slack form, while likely increasing the value of the objective function, and ensuring that the value does not **decrease**
- Repeat until the optimal solution becomes apparent

Example. Considering following example:
Maximum: $3x_1 + x_2 + x_3$
Subject to:

$$\begin{aligned} x_1 + x_2 + 3x_3 &\leq 30 \\ 2x_1 + 2x_2 + 5x_3 &\leq 24 \\ 4x_1 + x_2 + 2x_3 &\leq 36 \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

simplex. First we will transform into slack form, to do this, we need to introduce new variables (equaling to the number of inequalities), they are "slack" we have.
The slack we have will be x_4, x_5, x_6.

$$z = 3x_1 + x_2 + 3x_3 \tag{6.6}$$
$$x_4 = 30 - x_1 - x_2 - 3x_3 \tag{6.7}$$
$$x_5 = 24 - 2x_1 - 2x_2 - 5x_3 \tag{6.8}$$
$$x_6 = 36 - 4x_1 - x_2 - 2x_3 \tag{6.9}$$

In the slack form, the original variables like x_1, x_2, x_3 are called *nonbasic* values, while the slack values x_4, x_5, x_6 are called *basic* values.

Basic Solution : Set all nonbasic variables to 0

We set all nonbasic variables on the right hand side to some feasible value, and compute the values of the basic values.

For example, we choose $x_1, x_2, x_3 = 0, 0, 0$, now we have a basic solution $(x_1, x_2, x_3, x_4, x_5, x_6) = (0, 0, 0, 30, 24, 36)$ satisfying our requirement.

Pivoting Step: now we choose a nonbasic variable x_e whose coefficient in the objective function is positive, increasing x_e as much as possible, and set x_e as basic.

For example, we can choose x_1 at this time. To avoid violating any constraints, according to (6.9):

$$x_1 = 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4}$$

Using this to rewrite other functions:

$$z = 27 + \frac{x_2}{4} + \frac{x_3}{2} - \frac{3x_6}{4} \tag{6.10}$$
$$x_1 = 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4} \tag{6.11}$$
$$x_4 = 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4} \tag{6.12}$$
$$x_5 = 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2} \tag{6.13}$$

The basic solution here is $(9, 0, 0, 21, 6, 0)$, and the objective value is 27.

Pivoting Step: If we choose x_6 to pivot again, this will actually decrease the objective function, so we can either choose x_2 or x_3 here.

(The later pivoting is omitted) ■

6.4 Integer (Linear) Programming

Integer Programming is a bigger category. In this context, I am talking about Integer Linear Programming or ILP. That is, the objective function and constraints (other than the integer constraints) are linear.

ILP is **NP-complete**.

Definition 6.4.1 (ILP). An ILP standard form is expressed as:

Maximize: $c^T\vec{x}$

Subject to:

$$A\vec{x} + s = b \tag{6.14}$$
$$s \geq 0 \tag{6.15}$$
$$\vec{x} \geq 0 \tag{6.16}$$
$$x \in \mathbb{Z} \tag{6.17}$$

Note. Why I care about ILP? Because *polyhedral compilation* can transform the problem of rearranging dynamic execution instances problem (in code generation) to an ILP problem.

Chapter 7

Fourier Transform

Chapter 8

Information Theory