Intorduction to Optimization

Scientific Computing

Stefan Abi-Karam

Summer 2023

Table of Contents

1. Foundations of Optimization

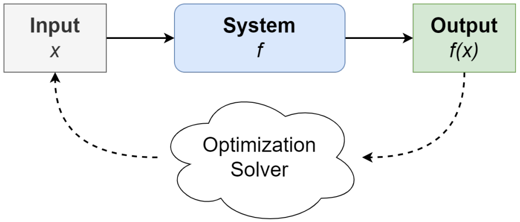
Table of Contents

1. Foundations of Optimization

Big Picture

Every optimization at a very general level can be viewed as the following:

- I have a set of inputs I can change: x
- I have a system I plug these inputs into: f
- The system has an output: $f(x)$



The goal of evey optimization problem is to find the inputs that give you the "best" output.

Big Picture

Nost optimization problems are formulated as **minimization** problems:

- I want to find the inputs that give me the smallest output: $x^* = \arg\min_x f(x)$

Sometimes I want the output to be as big as possible or as close to a target value as possible. We just reformulate these as minimization problems:

- I want to find the inputs that give me the largest output: $x^* = \arg\min_x -f(x)$
- I want to find the inputs that give me the output closest to a target value T: $x^* = \arg\min_x \mathrm{L}(f(x), T)$, example: $x^* = \arg\min_x |f(x) - T|$

Big Picture

We can also include **constainst** on our input variables that match with real-world descriptions of the problem. A problem can have no constraints, one constraint, or multiple constraints of different types.

Big Picture

Inequality Constraints: input variables are constaned by $<$, $>$, $\leq$, and $\geq$ using inequalities.

$$5.99x_1 + 2.99x_2 \geq 10 \tag{1}$$
$$0 \geq 10 - 5.99x_1 - 2.99x_2 \tag{2}$$

Real Example: The minimum order price for ordering parts is 10.00, I order x_1 units of item 1 and x_2 units of item 2 is .

Equality Constraints: input variables are constrained by $=$ using equations

$$x_1x_2 = 70 \tag{3}$$
$$0 = 70 - x_1x_2 \tag{4}$$

Real Example: The area of my rectangle that has width x_1 and height x_2 must be equal to 1.

Big Picture

Numeric Type Constraints: input variables must be of a certain numeric type

$$x_1 \in \{0, 1\} \tag{5}$$

$$x_2 \in \mathbb{Z} \text{ (Integers)} \tag{6}$$

$$x_3 \in \mathbb{R} \text{ (Real Numbers)} \tag{7}$$

Big Picture

I may also have multiple outputs that I want to minimize at the same time, known as **multi-objective optimization**:

- I want to find the inputs that give me the smallest outputs: $x^* = \arg\min_x f_1(x), f_2(x), \ldots, f_n(x)$

Multi-objective optimization is a bit more complex since you may have to figure out how to balance which output functions are more important than others. As you change the inputs, one output may improve while another worsens. There are more advanced techniques that solve this issue in different ways.

One simple example is to just minimize the weighted sum of all the outputs where you pick the weights yourself:

$$f(x) = w_1 f_1(x) + w_2 f_2(x) + \cdots + w_n f_n(x) \tag{8}$$

Big Picture

There are many labels you can use to descibe your optimizaton problem:

- Linear / Linear Programming
- Non-Linear Optimization
- Combinatorial Optimzaiton
- Mixed-Integer Optimization
- SAT: Boolean Satifiabiley Problem
- Multi-Obejecttive Optimization
- Balckbox Optimization
- Diffeienable Optimzation

Real World Problems

To apply optmization to a real-world problem, we must first **"translate"** the real-world problem to to optmization problem.

All optimzation problems can be described using a common "mathmatical description" we define later.

One weird thing is that there are multiple ways to translate the same problem into different "mathmatical descriptions" that are all valid. However some fomulations of the same problem are easier to solve and analyze than other formulations, even for computer solvers.

Real World Problems

Problem: I have x_1 pounds of flour and x_2 pounds of sugar. I want to maximize the profit I make from selling these material. I set the price of flour to 5.99 per pound and the price of sugar to 2.99 per pound. The wholesale cost of flour is 2.00 and the wholesale cost of flour is 1.00. I have a total of 10.00 to spend on these items. Assume I can sell all the items I buy.

Formulation:

$$\max f(x_1, x_2) = 5.99x_1 + 2.99x_2 - 2x_1 - x_2 \tag{9}$$

$$\text{subject to} \tag{10}$$

$$0 \geq 10 - 5.99x_1 - 2.99x_2 \tag{11}$$

$$0 \leq x_1, x_2 \tag{12}$$

This is a simple linear optimization problem.

Real World Problems

Problem: You are planning a field trip for your class and need to assign hotel roommates for the trip. You have n students (assume and even number of students) and two studetns will share one room, so you need $n/2$ rooms. Each student has ranked evey other student in order of preference for roomates. As a good enough solution, you need to assign the pairs of people so the final solution is "stable": there are no two people which are not roommates and which both prefer each other to their roommate as opposed to their assigned match.

Formulation: There is more than one way to formulate this problem as an optimization, but this problem actually has a deterministic solution that doesnt need an optimization algorithm to solve. There is also no guarentee that there will always be a "stable" solution.

This is a combinatorial optimization problem.