Optimization Algorithms: Theory and Software and Implementation

Prof. Thirumulanathan D

Department of Mathematics

Institute of IIT Kanpur

Lecture: 1

Hello everyone. Welcome to this course on Optimization Algorithms, Theory and Software Implementation. This is the first lecture. In the first two lectures, we will rapidly cover the basics of optimization theory. I will start with certain applications of optimization to have a head start. When we start with optimization, one always has this particular question: why optimization? What are the applications of optimization?

I am not sure if you would be surprised when I say that optimization is actually being used in everyday life by every person you can see. Ok, why do I say so? Consider the following problem, which I call the **problem of time management**. Ok, so any family man has a lot of things to do on a day-to-day basis. So, he has to do it every day.

He has to work in the office for about eight hours. He has to attend a meeting for about one or two hours. So, other than that, he has to run around to buy groceries and medicines for his family members. He has to prepare for the next day's meetings. He also has to spend certain time with his family and he also needs to have his food, do his other daily chores, and also have a sound sleep for at least six hours.

So that is quite a bit of things to do on a day-to-day basis. Right But to finish all this, he only has 24 hours. Right So, he cannot ask, saying that he has a very hectic day, "give me more time." So, he has to split 24 hours in such a way that he completes all of these tasks and also maximizes his happiness out of whatever he does. Right So let me actually model this as an optimization problem.

So, as an example, let me say that:

t_1: time spent on office work (in hours),

t_2: time spent on meetings,

t_3: time spent with family,

.

.

t_n: time spentsleeping.

Let:

$u_1(t_1)$: utility from time spent on office work,

$u_2(t_2)$: utility from time spent on meetings,

.
.

$u_n(t_n)$: utility from sleep.

The objective is to maximize the total utility:

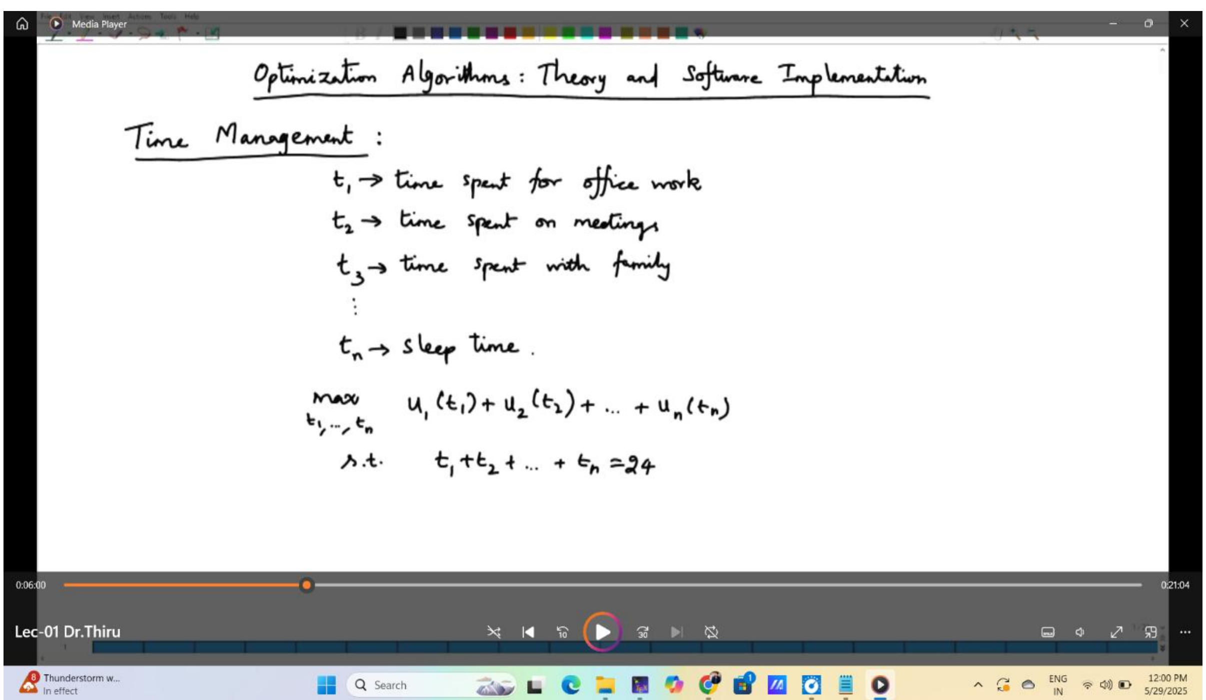
$$\text{maximize: } u_1(t_1) + u_2(t_2) + \cdots + u_n(t_n) \text{ Subject t}$$

the time constraint:

$$t_1 + t_2 + \cdots + t_n = 24$$

So, this maximization is over t_1 to t_n. You can see that this is actually an optimization problem where he has to maximize the sum of utilities, subject to the constraint that he has exactly 24 hours in a day. So, every one of us who is managing time on a day-to-day basis is actually solving this particular time management problem.

(Refer slide time 6:00)



So, that is why I said optimization problems are actually being solved by every person on a day-to-day basis.

You might actually ask me whether someone actually thinks about what his utility is here, what is his utility there, and so on. He may not do it every day thinking, I mean calculating the utilities exactly and finding the optimal amount of time and so on. But if you see what are his activities over, let us say, a few months or a year, you can actually see that he would have spent more amount of time for the task that gives him the highest utility and the next amount of time for the task that gives him the next highest utility, and so on.

This is a problem that every person is actually solving so that he actually manages his time effectively every day. So fine, that is a more layman kind of an example. We will look at some examples that are used in academics.

One of the key problems that economists solve is what we call the **utility maximization problem**. Let us consider the utility maximization problem that is well known in economics. So, we have a consumer having an income of W. Let us assume that he wants to buy many items as much as possible with his income.
So, let us say there are n items:

x_1 is the quantity of item 1 that he buys,

x_2 is the quantity of item 2 that he likes to buy,

.

x_n is the quantity of item n.

So, he actually gets a utility from $x_1, x_2, \dots, x_n$ when he buys a quantity of x_1 amount of item 1, x_2 amount of item 2, and so on. So, he buys $x_1, x_2, \dots, x_n$ in such a way that the budget constraint is satisfied and the other constraint that needs to be satisfied is that every quantity needs to be non-negative.
Let $P_1, P_2, \dots, P_n$ refer to the price of item 1, item 2, . . . , item n. This, as you can see, is another popular optimization problem where the consumer maximizes his utility subject to the budget constraint and the non-negativity constraint on the quantity of the items.
Another popular optimization problem is something in data science. Some of you might have heard of the term " **support vector machine**." In short, we call it SVM.
I am not going to elaborate on what is a support vector machine, but I will tell you what we are expected to do in a support vector machine. Ok, so let us consider that we have a training dataset which is represented as $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, where
$x_1, x_2, \dots, x_n$ represents the features, and $y_1, y_2, \dots, y_n$ represent the labels. So, that is for each data point, x_1 is a data point for which there is a particular label, x_2 is a data point for which there is a particular label, …, x_n is a data point for which there is another particular label.
So, we consider a binary classification problem. So, y_i is either −1 or +1. Given this setup, what do we want to do? We want to solve the following optimization problem:

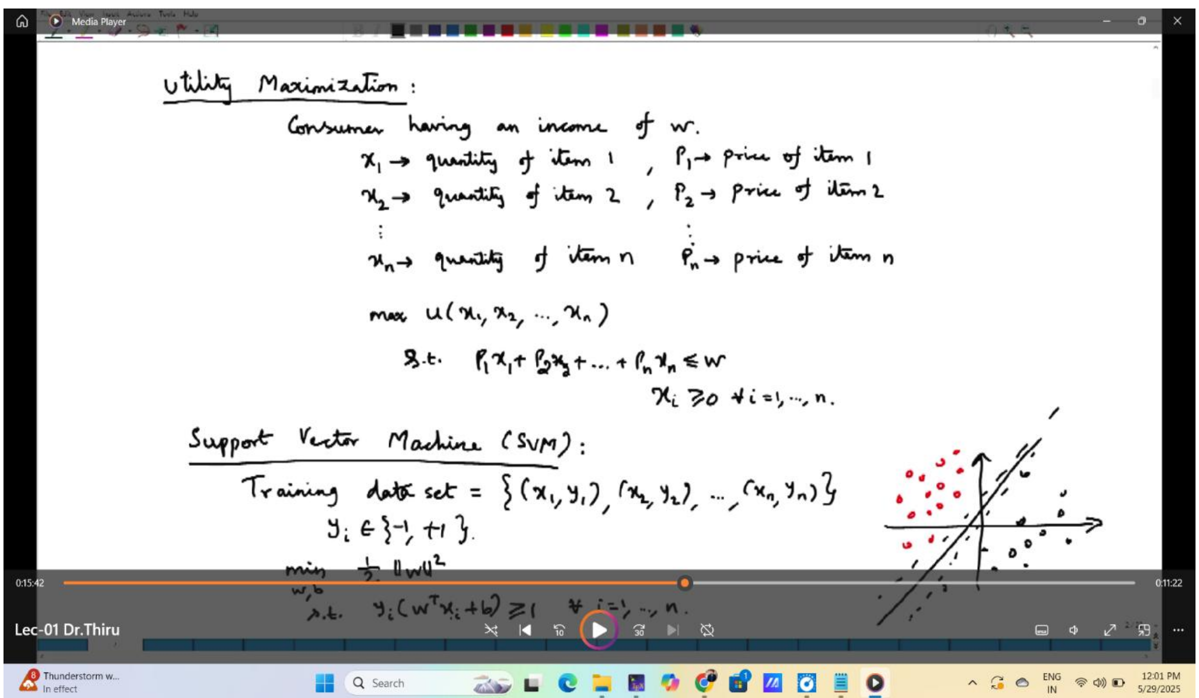
$$\min \frac{1}{2}\|w\|^2, \quad \text{subject to} \quad y_i(w^T x_i + b) \geq 1, \quad \text{for all} \quad i = 1, \dots, n.$$

Just to give you an understanding of what this optimization problem does: when you are given a separable dataset, for example, let us say, $x_1, x_2, \ldots, x_n$ are in two dimensions, features in two dimensions. So, let us say, I will draw those that are labeled as -1 in red. So, let us say these are the points and those that are labeled as $+1$, I will draw them in black. So, what SVM does is it actually separates, I mean, finds the best separator between the set of points that are labeled as -1 and the set of points that are labeled as $+1$.

So, note that there are many possible lines that can be drawn. For example, we can draw this line. That is one possibility. We can draw this line, that is another possibility. We can draw this line, that is another possibility. So, just by looking at the figure, you see that somehow the one that I drew in thick black right seems to be the best separator rather than the others. Right So, such a best separator is actually found using support vector machines by solving this optimization problem.

Now, we have seen examples from day-to-day life, from any layman's life, and also from those that involve academic interests like economic sciences and also data analysis.

(Refer slide time 16:15)



If you see the problems, they are of the following form. In each of the problems, I am actually minimizing some function $f(x)$. So, let me write it as:

$$f(x_1, \ldots, x_n) \quad \text{subject to the condition that} \quad x = (x_1, \ldots, x_n) \in S.$$

Whenever we have a problem of this sort, we call it an optimization problem. You can see that f is some function from R^n to R, and throughout the course, we assume that this function is twice differentiable.

Given that we have got the applications right at this point in time, we will go into the theory as fast as possible. So, as far as optimization is concerned, there is a broad classification which we call unconstrained optimization and constrained optimization. So, optimization can be broadly classified as **unconstrained and constrained**.

To explain mathematically what is the difference, we call the problem unconstrained if $S = R^n$. So, this set that we have defined here: x_1 can take any real value, x_2 can take any real value, x_n can take any real value. Then, I call such an optimization problem as unconstrained. If $S \neq R^n$, then there is a constraint on what values $x_1, \ldots, x_n$ can take, and such an optimization problem is called a constrained optimization problem.

As far as algorithms are concerned, again we would separate them as algorithms for unconstrained optimization and algorithms for constrained optimization. So, that is a very broad classification.

So, we will now understand more about the solutions of an optimization problem. Let us consider this problem of minimizing $f(x)$ subject to $x \in S$. When do I call a particular point x^* as a solution?

So, again in solution, there are two kinds of solution concepts. A solution can either be a global solution or a local solution. So, what is the definition of a global solution? x^* is a global solution of the above optimization problem if:

$$f(x^*) \leq f(y) \quad \text{for all} \quad y \in S.$$

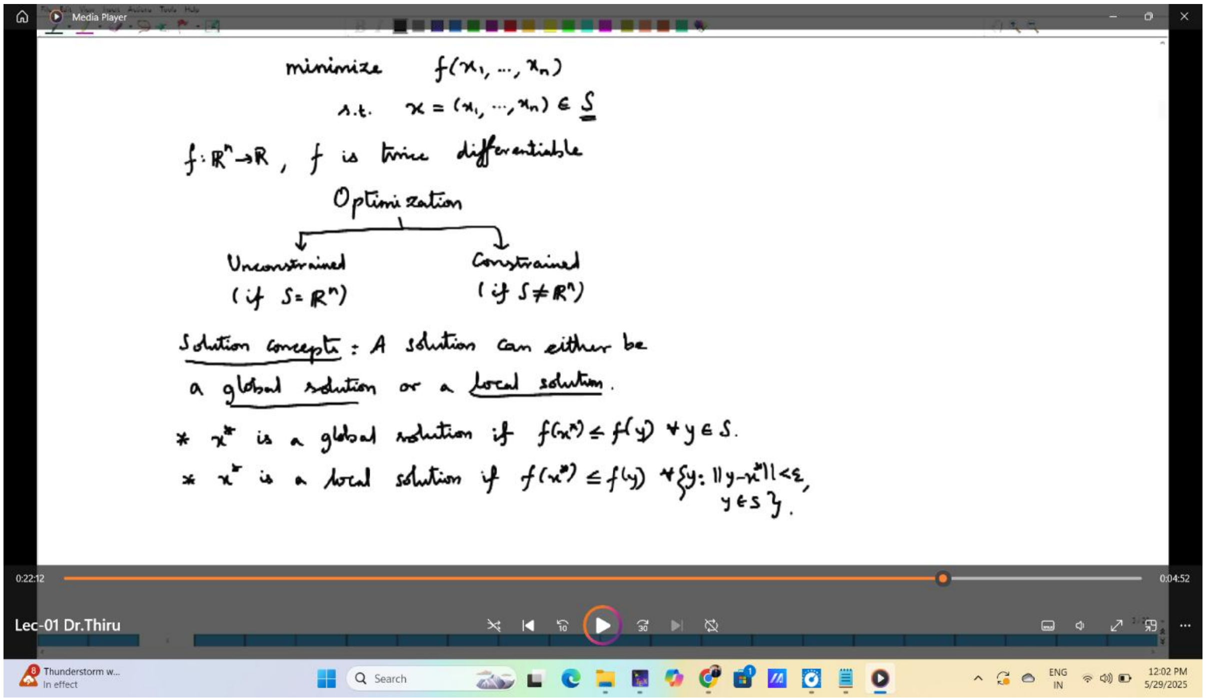
So, this is when we call x^* as a global solution. It is a local solution if:

$$f(x^*) \leq f(y) \quad \text{for all} \quad y \quad \text{such that} \quad \|x^* - y\| \leq \epsilon, \quad \text{for some small } \epsilon.$$

So, what this means is that if you look at a ball around x^* and you verify whether within that ball, $f(x^*)$ is less than or equal to $f(y)$, the ball must also intersect with the set S.

It is easy to see that every global solution should also be a local solution. But the reverse is not true. This means local solutions are a weaker concept and global solutions are a stronger condition that needs to be satisfied.

(Refer Slide time 22:16)



Let me give you some examples to clarify global and local solutions. We can have optimization problems where there are no global solutions or no local solutions. For example, consider $f(x) = x$. If we are trying to solve the problem:

$$\min f(x) \quad \text{subject to} \quad x \in \mathbb{R},$$

You can see that the minimum does not exist. In layman's language, the minimum is at minus infinity, but minus infinity is not a part of R. Therefore, we say that a minimizer, a solution, does not exist.

Here is another example where a local solution exists but a global solution does not exist. Let us consider:

$$f(x) = (x-1)(x)(x+1),$$

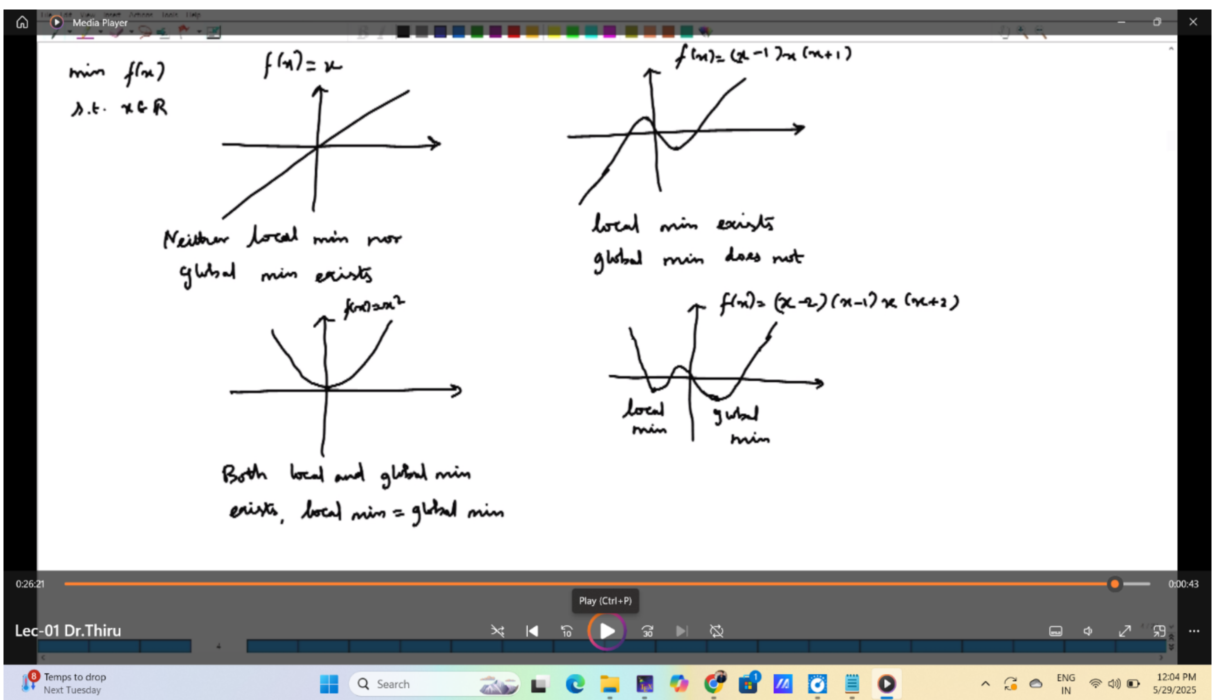
which is plotted as follows.

We can see that there is no global minimum, but we have a local minimum at $x = \frac{1}{2}$.
So, this is a case where a local minimum exists, but a global minimum does not.
Finally, let us take another example where both local and global minima exist. So, the nicest example is where $x = 0$ is both the local and global minimum. Both local and global minima exist, and local minimum equals global minimum.
Lastly, let us consider an example where both local minima and global minima exist, but you have a local minimum which is not a global minimum. Consider the function plotted as:

$$f(x) = x^4 - 3x^2 + 2,$$

where there are two local minima. One of them is the global minimum, but the other is only a local minimum. This is only a local minimum. So, these are certain examples to illustrate the differences between the local minimum and the global minimum.

(Refer slide time 26:25)



In the next lecture, we will actually look at conditions where the local minimum and the global minimum will be one and the same.

Thank you.