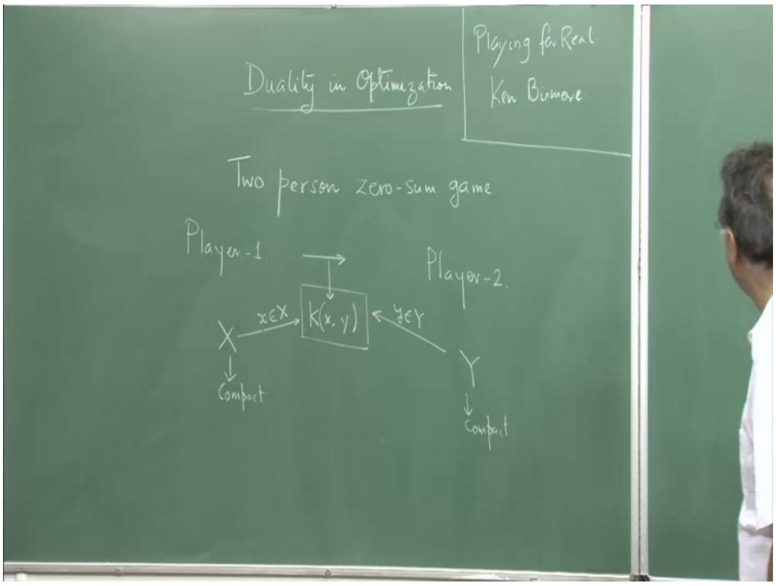
Foundation of Optimization

Prof. Dr. Joydeep Dutta
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture - 32

Today, we are going to talk about duality in optimization. We had already said that, we would talk about duality.

(Refer Slide Time 00:27)



Let us just write down the topic of our discussion today. Duality is a vast topic subject by itself as a part of optimization. The lots and lots of strategy (( )) to duality. What duality says that, behind every minimization problem, there is some maximization problem, which is simultaneously going on; and there is a relation between the solution values of the original minimization problem and this dual problem, which is going on at the back. So, the maximization problem, which is going on at the back is usually known as the dual problem. And the original problem that you want to solve the minimization problem that we are using here, is called the primal problem.

Now, these issues of two problems are being handled simultaneously. Though actually handing one problem at the back, automatically there is another problem getting tackled, is an issue of very big significance, because what would happen is that if original problem is difficult to handle, it is in many cases for example, in the case of linear programming, the dual problem is simpler to handle. An important issue is that are the solution values of the primal and dual problem equal; that is a very, very fundamental question, which we will discuss.

Of course, we do not have the courses coming towards an end and we really do not have so much time to discuss duality in detail; otherwise, duality itself would be 10 or 12 lectures. So because this is the foundational course we are trying to pack in many things into it and trying to give you as much as rather some sort of a broad picture of the subject rather than going into details of everything.

Now, this issue of duality is linked to what is called two-person zero-sum game. We will use this idea of two-person zero-sum games – a concept, which was very well-studied by Voronoi and Morganstern in their famous book game theory and economic behavior. So, this two-person zero-sum game means there are two people who are playing a game. So, there is a player 1 and there a player 2. There is a player 1 and there is a player 2. Player 1 has a strategy set X say in R n; we call this set let be compact and convex or whatever; maybe compact is fine. And, player 2 has a strategy set. For example, you are playing chess; you have your strategy set and other person has their strategy set; nobody knows what strategy you have. So, these strategies are not revealed to each other. These also assumed to be compact. So, player 1 can take an x belonging to X and make one move in the game; while corresponding to it, player 2 can make a move by taking y from Y.

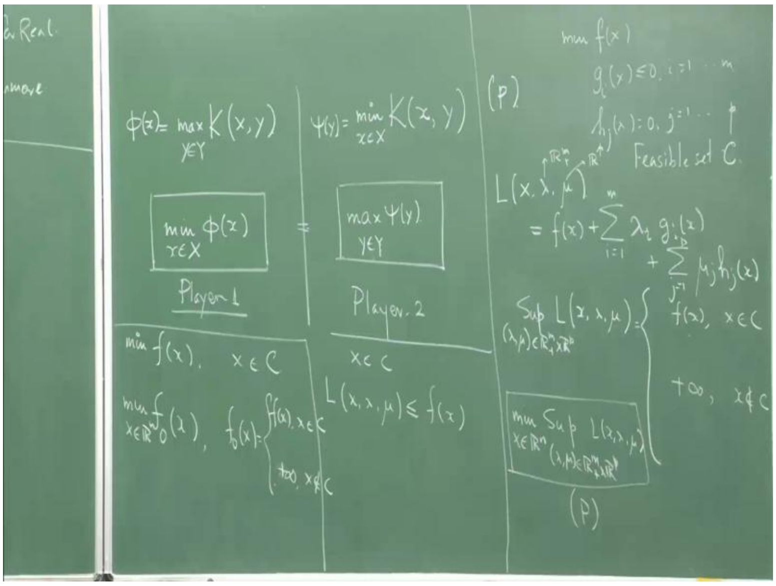
And, whenever there is such a move having the strategy pairs x and y, player one plays player 2 and amount K x, y rupees say. So, this is usually called the payoff of the game. Now, why payoff is there? Why you need to define, is a big big issue. I would rather for those who would like to go more into games, it would be very beneficially if you look into this book – Playing for Real, Ken Binmore. This is a fabulous book on game theory; my favorite actually. That is why I also want to share with you. That tells you why you need to payoff, how do you design a payoff. So, there is lot of issues of designing payoffs and all those things. So, they have issues related to the economics. It is pretty interesting by the way. But let us just take that. He is making a…

Now, it does not mean that every time this guy – player 1 is giving player 2 money. If he is giving money – player 1 to player 2, then K x, y is positive. If he is receiving money, then K x is… If he is giving money, then K x, y is positive. So, he is giving away the money.

And, when he is receiving money, it is negative; rather when K x… because he receives money player 2, it is positive for him. And, if he gives away money, it is negative for him. So, that is the way it is.

So, every time this amount – certain amount is been given. So, whatever is his gain is his loss. So, the total payoff, the total gain of the players is actually 0. So, that is why it is called the zero-sum game. If player two is receiving K x, y amount of money, then player 1 is losing K x, y amount of money. Now, let us look into (( )) the way player 1 and player 2 would actually think when they are playing the game. Now, player 1 knows that he has to pay some money – K x, y.

(Refer Slide Time 07:01)



He wants to know that, if he pushes this strategy X; if he plays a strategy X, he has no idea what strategy from Y would be known. So, their strategies are not revealed before and what I will play. So, for the time being let us know that, both of them know their strategy sets. For example, when you play chess, you know that you… Suppose you are super X person, you would know that, these are the only possible moves you can make in the game. So, in that way, both possibly know the strategy sets; both could be x actually then. That is a different thing. So, here just for generalization, we are writing x, y and all those things.

Now, give him this strategy X. Once he had played x, he has no idea what move, why the player 2 is going to give. It could be any y in the set capital Y. So, in that way, given… So, he wants to know what is his maximum loss, which is very natural. If I am also playing, I want to know K; if I play X, what is my maximum loss? So, I want to maximize… Obviously, Y would like to… When I play, 2 would like to put a move, which would take as much money as he can from player 1.

So, player 2 would like to make a move, so that as much as money he can draw from player 1, he would like to take. So, player 1 wants to know beforehand, what is the maximum loss he can have; what is the maximum loss of money he can have once he has played the strategy X; which means… If I call this a phi x; for every move x in X, he is noting the maximum loss. So, anyone would try to minimize this maximum loss. If anyone is player 1, his natural idea would be to plan an x for which this is minimum. So, this is the minimization problem. So, player 1 is actually solving a minimization problem. So, K x, y of course, is a function of x and y, which gives you a real number, because it is rupees. Now, this is what player 1 is playing.

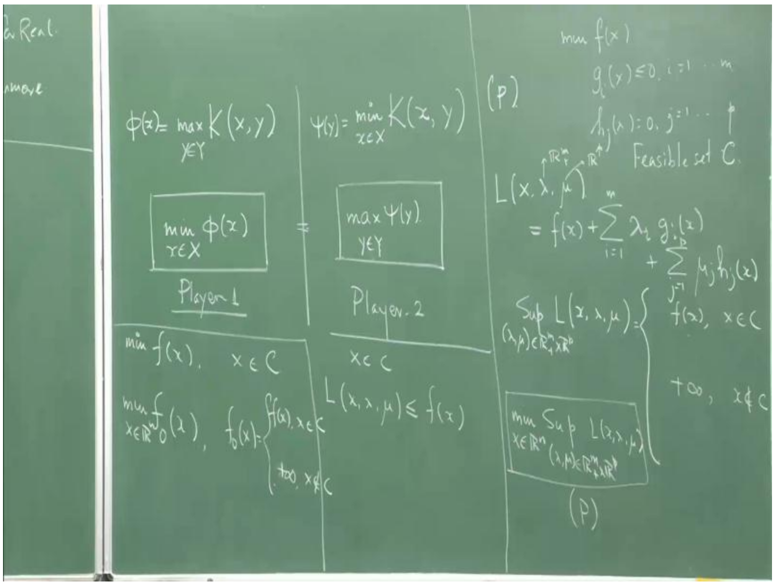
Let us look at what player 2 is thinking. So, player 2 now… Suppose he has given a strategy Y and he does not know. For example, corresponding to his strategy Y, what move the player x is going to make? Once he has made his move y, he obviously does not know what move the player 1 is going to make. Now, player 1 can make any move x from his set x. Important part is that, this guy obviously knows that, he is going to get a money K x, y amount for every x he moves.

But he also believes that he does not know… He also wants to know, what is the minimum amount he can extract from player 1? So, he would like to see for a given y, what is the minimum amount of money or the minimum amount of profit he can make from player 1. So, this is the minimum amount of profit that player 2 is going to make. So, if I am player 2, I would like to maximize that minimum profit; everybody would like to maximize profit. So, I would like to maximize phi y. You can of course write this player 2’s problem. So, player 2…

As player 1 is a solving minimization problem, player 2 solves this maximization problem. Technically, you can say that, I can write this whole problem as mean of max of this and this as max of mean of this. Of course, you can do that. The question is now, when are these two values equal? If these two values are equal, then the x for which this value is equal to this and y for which this value equal to this, is the optimal strategy of the players;

that is, if I deviate from that strategy, neither of us are better off. So, the very important question is, when is these two values equal under what condition on K possibly? What should be the behavior of K, which will make these two values equal. So, this is your… So, player 1 plays the primal problem; player 2 plays the dual problem.

(Refer Slide Time 07:02)



When it comes to the standard mathematical programming problem, which we had been writing as p, that is, to minimize effects subject to… Now, can you design something like this? Is this problem – can it be represented as a mean max problem – mean of max of some function depending on two variables? Can it be written like that? Is it possible? That is the fundamental question.

Basically, you are trying to imitate this whole process that we have described just now for a standard mathematical programming problem. Now, the question is if that is so; if you want to do so, you want know how, what it would be a payoff function; what is the analog of K x, y that you know for the games; what is the analog for the mathematical programming problem. Here the analog is the Lagrangian function, which is written as f x plus… Now, in this case, lambda is in R n plus and this is in R p; x of course, is in R m. Now, this is a standard Lagrangian function, which we already know.

Now, can I write this original problem as max of L x lambda mu over mean of something. Let us see what would happen if I want to… So, in this case, lambda and mu are assumed to be the strategies of the player 1 – the original problem. So, here let me see what would happen if I do this. So, varying lambda mu on R n plus and R p – R n plus R p, let us see what would happen to this. Now, let me call the feasible set of this problem p feasible set as C, which consists of all x, which satisfies g i x less than equal to 0 for every i and h g x is equal to 0 for every j from 1 to p.

Now, once you know that C is a feasible set, now consider an x in C. Then, of course, all the h j x is equal to 0; g i x's are less than equal to 0. So, then, whole of the thing – this whole Lagrangian, when you are taking x is in C, then the Lagrangian for any lambda mu you choose in this set; that is less than equal to f x. But, if you choose lambda and mu – both to be zero, zero vectors, then that is exactly equal to f x.

So, the supremum or maximum; or, the supremum I would say, because you cannot use the word maximum in that sense, but you can use the word supremum. Here I use the word maximum or minimum, because I know these are compact and assuming that these are continuously in I, when i is a continuous function. So, here this is f x if x is in C. So, what is the answer if x is not in C? If x is not in C, the supremum blows up; it is plus infinity. So, how it is plus infinity? If x is not in C, there means either there is an h x x; there is a j for which h j x is not equal to 0 or a i or both in fact; or a i for g i x is strictly bigger than 0 or both.

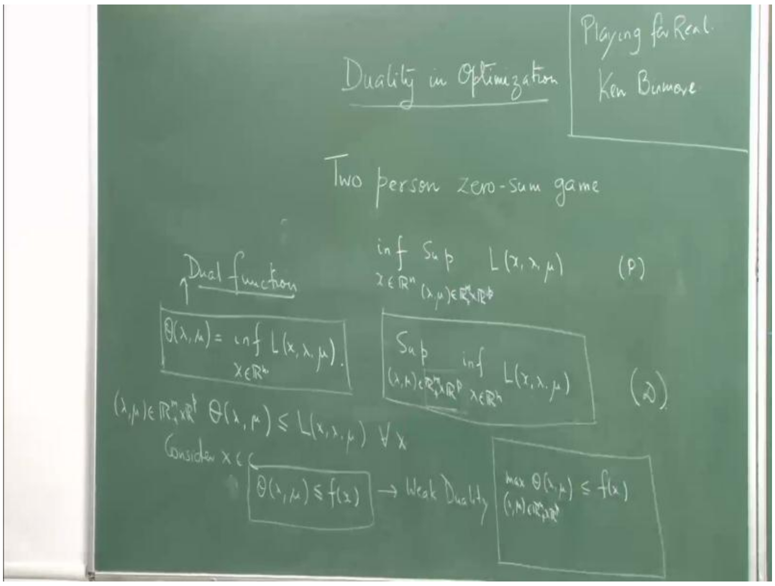
Suppose I just take one of the cases – take an x, for which h j x is not equal to 0. Without loss of generality, assume h j x is positive. So, make all the lambda mu's to be 0; and, keeping the mu associated with that particular h j, that is, mu j to be positive. Now, you keep on increasing the mu j as much as you want. Then, moving along that mu j, as you increase mu j, the value of the Lagrangian function will keep on increasing and it would not decrease. And, I can so blow up the value. And, that is why you have this.

Now, this is slightly different from look of this. So, if x is in C, you will get this; if this is this, you will get this. This is a structure; this can also be written as something else. This is a structure, which is used very often in modern optimization, which is actually borrowed from this sort of, this way of writing the original problem p, representing the original problem p. So, if you want to minimize a function f x over a set x element of C, which is a constrained problem, you can convert this constrained problem to an unconstrained problem. You see you are actually con…

Now, what I have to do is to take the minimum. So, for the original problem now, is to take the minimum or infimum if you want to write. So, my original problem p can be now represented like this. So, whenever x is element of C, this is f x or else it is plus infinity. So, in many… So, this problem can now be written… The same structure can be written putting the simpler form. That I can now write this whole problem as minimize f naught x, where x is in R n; where, f naught x is f x when x is element of C; and, is plus infinity when x is not element of C.

So, this is… Basically, I am putting infinite penalty for not maintaining feasibility and that this conversion of a minimization problem of finite valued functional over a set C can be converted into the minimization of an extended real valued function over the whole of R n. And, this structure is the hallmark of modern optimization and it actually comes from this formalism – this way of writing the original problem in terms of the Lagrangian function. So, this is my player 1's problem. So, how do you write the player 2's problems? That is the whole question. So, the original problem can be written like this. So, could it be that i can change this to… So, my dual problem.

(Refer Slide Time 21:56)

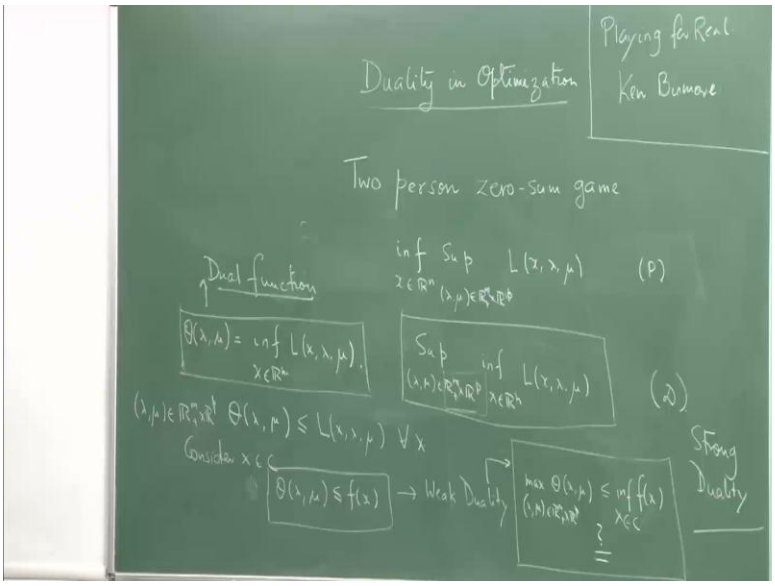


If my primal problem p is say inf or maybe become more erudite and write… So, if this is my original primal problem p, is my dual problem can be done by swapping these two? That is, sup of… In fact, indeed if that was the player 1's problem, the primal problem, this is indeed the dual problem. So, the dual objective function, which is phi y in the game, now, can be written as theta lambda mu, which is infimum… So, this is called the dual objective function or the dual function. Now, the question is why is the dual problem important? Fine, you have found that there is another problem, which is going on at the back of the primal problem and then then, but but… So, what is this?

Now, consider any fix of lambda and mu; where, lambda mu belongs to… And then, consider this function. By definition, for whatever x you take, this is… Now, suppose this is true for all x; consider any x in C. Consider x to be in C. Then, theta lambda mu is less than equal to f x. So, if this my dual function, then the dual feasible set is this – R m plus cross R p. This is a dual feasible set. Now, here you see what we have done is that, we are asking what would happen if x is in C. But when x is in C, each of h j x is 0; lambda i g i x is less than equal to 0.

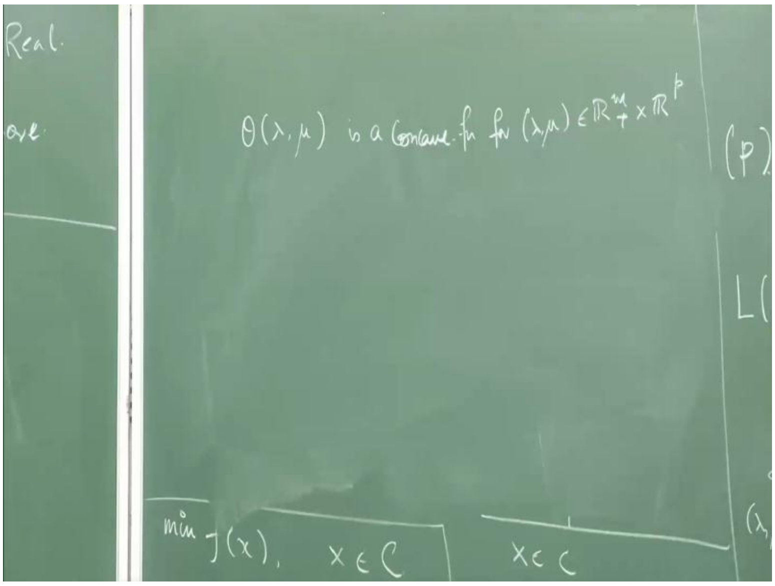
Finally, that whole thing is less than f x. So, for any pair from the dual feasible set lambda mu is taken and any x from C is taken, this is always holding. So, this result is called the weak duality. Now, this can be posed in a better way. Now, what does it tell you is a maximum of theta lambda mu. So, if this is holding for theta lambda mu; for every lambda mu, if I fix the x, then this whole thing would be less than equal to f x for a fixed x; where, lambda mu belongs to…

(Refer Slide Time 26:48)



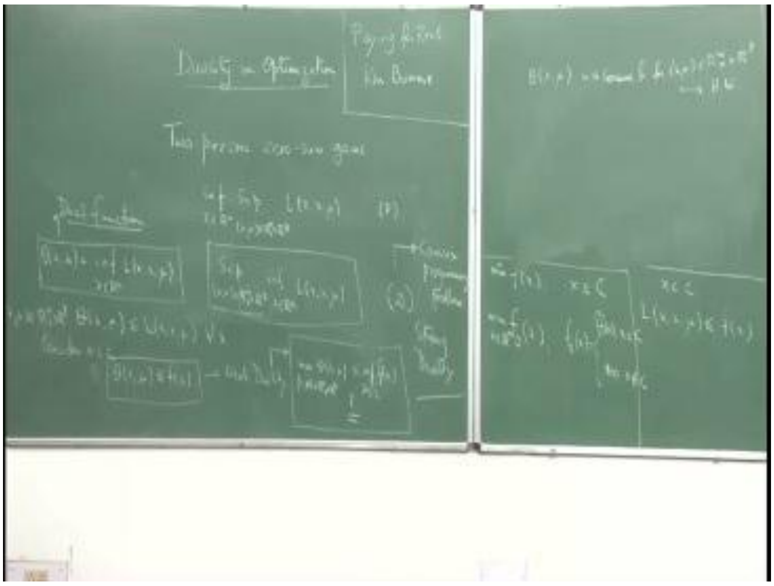
Now, once this is done, this is true for whatever x you choose, fix up; which means I can again take an infimum of the x's, which is my primal problem over C and i will get this relation. So, this is the consequence of the weak duality inequality. So, this is sometimes also called the weak duality; either this or this, whatever you want to call, because this is as same as this; this implies this. Now, once this is known; this weak duality is known, the question is, when will there be equality? So, when… if equality holds, that would result into what is called strong duality. Strong duality is something what we need. But, before we go in and discuss more about strong duality, we would like to get into some examples, etcetera and we will start with some features of the dual function itself. But these features – some of these features would be given to you as homework.

(Refer Slide Time 28:04)



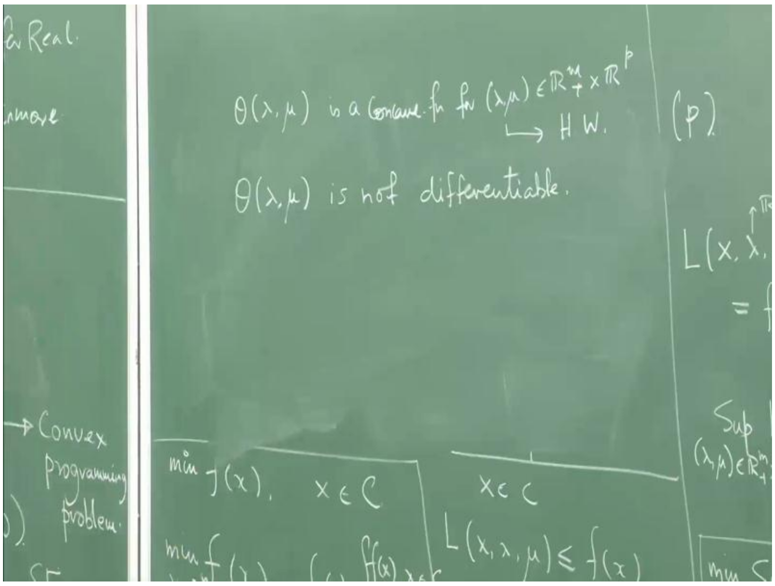
Theta lambda mu is a convex function for lambda mu – not convex, I made a mistake; concave function. Please correct this; it is a concave function. I should have written minus lambda mu theta. So, what you are doing, you are maximizing a concave function; or, in the same sense, minimizing a convex function. So, every local maximum of the concave function is a global maximum. And hence, the dual problem…

(Refer Slide Time 28:58)



This problem D is a convex programming problem. So, this problem is a convex programming problem. Now… So, here comes the first advantage of duality. Now, this is something… This (( )) You take it as a homework; HW is shorthand for homework as always. Now, what I want to stress is that, here comes the first advantage of thinking about the dual problem by taking the dual problem into account. Is that we have a convex optimization problem, which is actually solving whose solution actually provides to me the lower bound to the solution of the original problem. See if I can solve, because convex optimization problems have well-known good solution methods; if I can solve the dual problem and get a finite value as the solution, then I know that, my original problem cannot have a minimum value beyond this if I immediately know my original problem has a solution. And, if the original problem now has (( )) a structure like a quadratic programming problem over a polyhedral set, then I immediately know that there would be a minimizer existing for that. So, these sort of advantages can be obtained by looking at the dual problem. But, the dual problem has one major disadvantage. The major disadvantage is as follows.

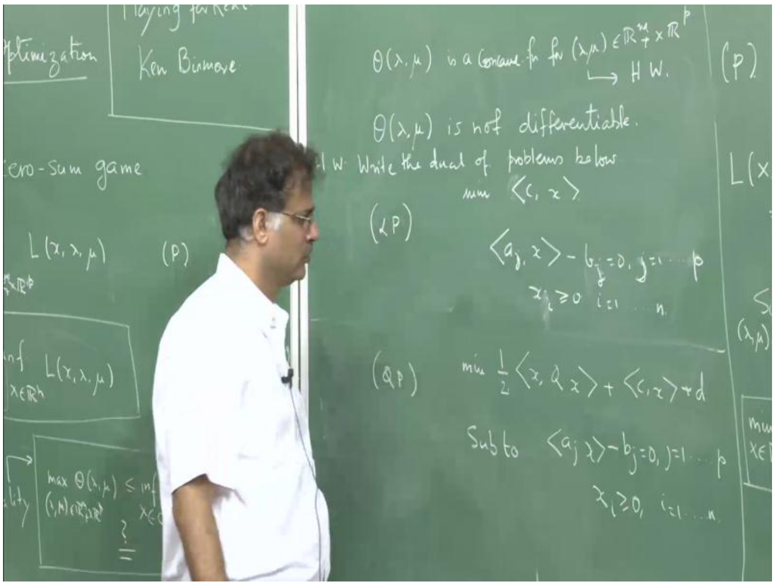
(Refer Slide Time 30:52)



This is not differentiable at every point. See this is very important to know that, this is not differentiable, because you are basically minimizing. How do you write down this one? So, Once you fix x in terms of lambda and mu, this is a linear function or an affine function actually if there is an added f x. So, these are affine functions. So, you are taking the minimum of a family of affine functions. You are taking the minimum, because each affine is convex, concave – both convex and concave actually. So, you are taking the minimum of this. And, what you are getting out is a concave function. So, here for every fixed x in terms of lambda and mu, this is a linear function. And now, as you vary this index x; that is, if I am making… So, that x is generating the family of concave functions. When you take the infimum, you will get a concave function. So, this is a concave function.

Now, this non-differentiability might turn out to be problematic. So, first, we will write down few examples, which as a homework, you should learn how to construct the dual problem. Once you learn to construct the dual problem, you will generate more confidence in the issue of duality. After we write them down, we go ahead and we describe a problem from where, from the geometry, you can see that the equality of these values are guaranteed; equality of these values are guaranteed. So, let us just write down the problem that I want to give you as homework.

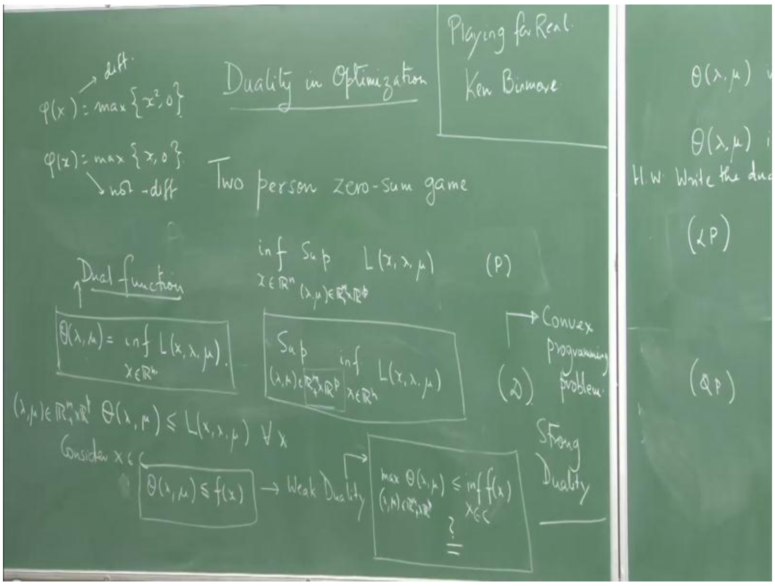
(Refer Slide Time 33:19)



So, let me start with this problem. So, take the linear programming problem L P, where you minimize the linear function over affine constraints. Maybe I will put h j. So, C is in R m and a j's are in R m. So, b j's are in R. Now, can you write down… So, the next problem is a quadratic programming problem, where I replace the linear function by a quadratic function with hessian matrix Q and which is positive semi-definite subject to the same constraints. Of course, I will give you these calculations in the FAQs. So, those who would not be able to calculate them out, will calculate them for you. So, you need to look at the FAQs. So, write down the dual problem. Homework is to write down the dual of the following problems. So, these dual is sometimes called the Lagrangian dual, because you are using the Lagrangian function.

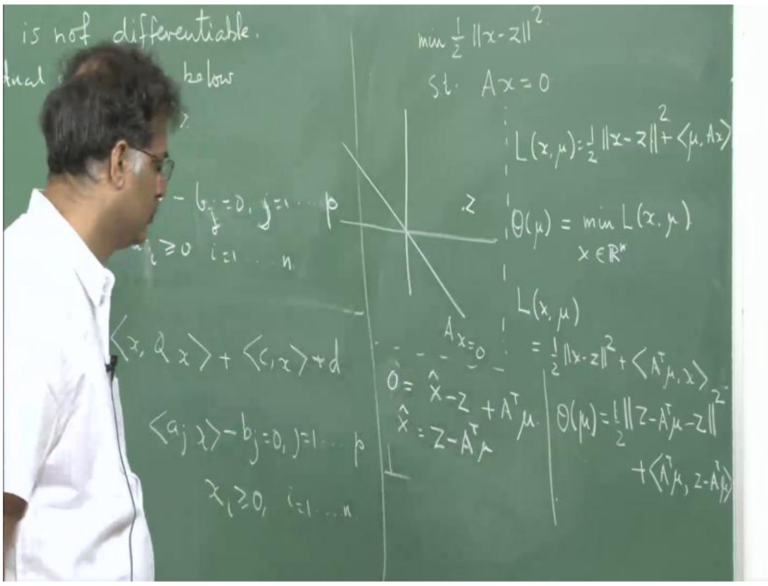
So, now, we shall show through an example, how the very geometry can show you the duality actually exists – strong duality. See what happens, if you just have equality constraints, then the dual problem, that is, taking the supremum of this; supremum is only taken over mu; then, the dual problem actually becomes an unconstraint problem, which is a very big advantage actually. We will also show… If time does not permit today, then tomorrow, then that I will show a case, where the dual function is differentiable and I will show a case, the dual problem is non-differentiable. Whenever we have the dual function written like this, does not always mean that the dual function would be non-differentiable, but in general, it is non-differentiable.

(Refer Slide Time 36:53)



For example, I will just give you a function. So, for example, if you take phi x is max of x square and 0, this is differentiable. But if you take phi x as max of x and 0, this is not differentiable. So, this is differentiable and this is not differentiable. So, theta lambda mu written like this does not tell you that, that is differentiable and that is always non-differentiable. It could be differentiable; it could be non-differentiable. Now, let me just try to write down a result, rather a problem and the very basic problem, the projection problem; and, let us see what you get out of it. So, you are fixing up a point z. Take a point z and take a matrix A say m cross n matrix to be very simple. (( )) m cross n matrix also, but let us just take m cross n matrix if you want. Or if you are comfortable with m cross n, we will go for m cross n.

(Refer Slide Time 38:17)



I am looking for a projection problem; that is, meaning I am trying to find the distance between z, where x belongs to kernel of A, where A is a m cross n matrix. So, this problem can be written as… So, it is a convex programming problem of course, because this is a convex… So, geometrically, A x equal to 0 is a subspace in R n. There is a set of all x, which satisfies A x is equal to 0, is a subspace in R n; there is a set of all x, which satisfies A x equal to 0 is a subspace in R n. And you already know that, kernel is a subspace. And, the subspace has to pass through 0. Suppose for the time being that, this is the line x equal to 0; I am just expanding into two dimensions.

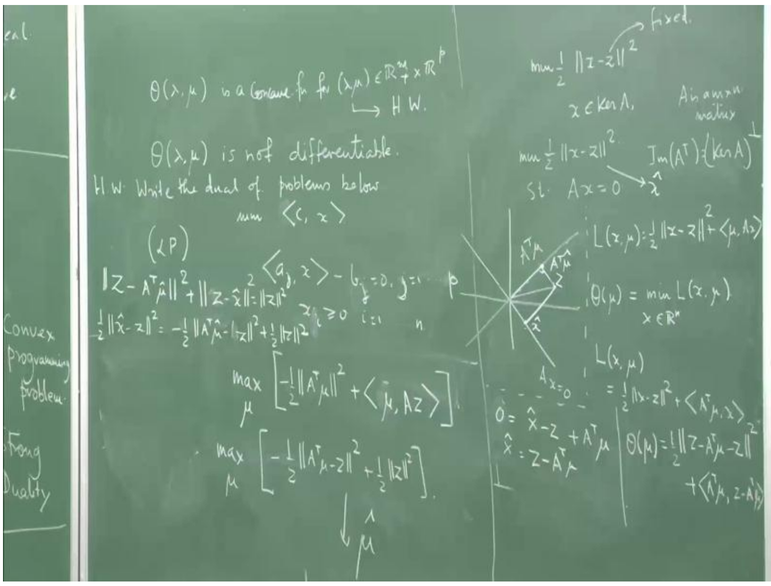
Now, what would happen if I try to write down the dual of this problem? The dual of this problem means we have to write down the Lagrangian, which is half of… See this z is fixed. So, z is fixed. It is from z, we are trying to find the… Basically, there is a fixed z here and we want to know the distance of set from this; that is, find a point on x upon x equal to 0, which is nearest to z.

Basically, you drop a perpendicularly from z to x and geometrically find what is that point, where the perpendicular will hit the line x equal to 0. Actually, when it will be in this sort of a thing, it will be just A x equal to 0; there is just A 1 x 1 plus A 2 x 2 equal to 0. But here I am just writing A x equal to 0 to keep the general flavor. This plus mu times A x; h x basically. So, mu… I am not writing summation mu; I am just writing like this, because if I write each one of them as A 1 x, A 2 x, A 3 x; then, I write mu A 1 x, mu 2 A 2 x. So, I am just writing it in a more compact form. I think this amount of mathematical maturity you should have towards the end of this course.

Now, if I want to find the dual problem, I want to construct theta mu; which means that, I need to minimize L x mu over x element of R n. But, this is a convex optimization problem; this is a convex problem. Why? Because you can write once you have fixed the mu, for fixed mu, L x mu is half of… Can be written as… So, in x, it is a convex function. And, for a convex function, every critical point is a global minimum.

Now, then if I differentiate this and equate it to 0, x would be equal to minus so solution of this problem say x hat would be equal to minus A transpose mu; not really; I am making a mistake; it should be… If I differentiate, then it will become x minus z plus A transpose… mu equal to 0. So, this is x hat. So, if I differentiate it, this is the derivative I am equating it to 0. And, the x for which this derivate would be equal to 0. Now, this would be equal to 0 – the x hat. For that x hat… That x hat is the minimum minimizer of this problem basically; which means x hat is now can be written as z minus A transpose mu. So, theta lambda mu… Theta mu, not lambda mu; theta mu, because there is no lambda; theta mu is now minimizer of this. So, how do I express that minimum? So, the minimum value is putting this in place of x. So, half of z minus A transpose mu minus z whole square plus A transpose mu into z minus A transpose mu. So, the origin now… which means that, the original problem…

(Refer Slide Time 44:28)



Now, the dual problem of this can be written as maximize over mu, which is an unconstrained problem. So, it can be written as maximize over mu. Now, if I do some manipulations here, because z z would cancel and you will have A transpose mu norm square from here also. So, that would give you minus half norm A transpose mu whole square plus A transpose mu z or mu A z. So, this is… Now, I have to maximize this over mu. know, I can add a half norm z square to complete the square and then I can write this as… So, this is equivalent to same as writing as… Basically, we add plus half norm z square and put minus half norm z square. So, you will have max of… over mu; mu is in of course, R m. So, if you are not sure, then let me write for you mu is in R m. And, this is what I will get as norm A transpose mu minus z whole square plus half norm z whole square. So, what is this measuring?

See if I take all the mu's in R m, then A transpose mu is nothing but the image space; A transpose mu is nothing but the image space of A transpose, which is actually orthogonal to the subspace A x equal to 0 to the kernel of the A. So, you know image space. This is image of A transpose. This is basic linear algebra, is the orthogonal complement to kernel of A. Using that idea, this is nothing but you are minimizing the distance of z from this thing – A transpose mu. So, basically, you drop here; you want this. So, find a mu hat such that A transpose mu hat would minimize this. So, this is the A transpose mu hat position. And then, corresponding to this, as you know for the convex case, these are close convex set; there would be an x hat; there would be a projection. Let this be the projection on x hat. Then, what does it show? This distance is A transpose mu and this distance is x hat. So, this distance is x hat minus z square; and, this distance is z minus A transpose mu. So, what is happening is, if you look at this distance, this is z minus A transpose mu. So, basically, you will have…

Finally, you will have z minus A transpose mu hat; this distance; square of this plus square of this, is nothing but by Pythagoras theorem, the norm of this whole square. So, here is mu hat, where the perpendicular lands; here it lands that x hat. So, you see now, this plus z minus x hat is equal to norm z square. So, finally, what I get from here is half of norm x hat minus z square is minus half of norm A transpose mu hat minus z whole square plus half norm z square. So, mu hat is a solution to the dual problem; mu hat is the solution to this problem. So, the solution to this problem is mu hat. And, solution to this problem is x hat. So, this is the maximum value of the dual problem and this is the minimum of the primal problem; and, they are equal.

Here you see we have not put any additional things on the constraint, which gives us a fact that, if I have linear or affine constraints for a convex programming problem, possibly, the strong duality always hold. The answer is yes; which we will discuss in the coming 2-3 class. We will give more examples and we will discuss in more detail the basics of strong duality and its applications to certain interesting classes of problems.

Thank you very much.