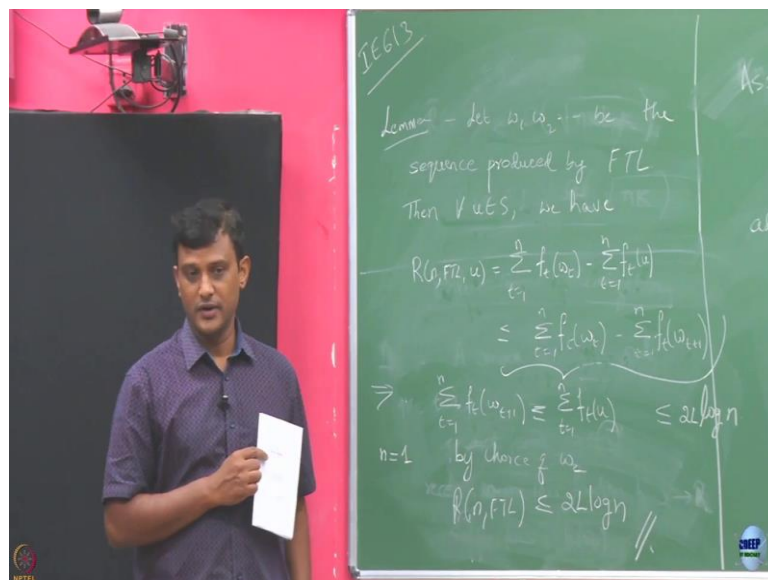


**Bandit Algorithm (Online Machine Learning)**  
**Prof. Manjesh Hanawal**  
**Industrial Engineering and Operations Research**  
**Indian Institute of Technology, Bombay**

**Lecture – 19**  
**Follow the Leader (FTL) Algorithm**

So, before we show the regret we get by this algorithm Follow the Leader. We are just show going to show that the regret whatever we had is can be bounded in terms of this loss function observed at two consecutive predictions.

(Refer Slide Time: 00:42)



So, just me make that formal that is lemma, and I am also going to write it as this is a FTL policy and also I am going to use a reference  $u$  this is given as  $f_t(u)$ . So, like when I looked earlier at the regret I had a  $\min_u$  over all  $u$  right, right now I am not looking at all  $u$ , but I am looking at one particular  $u$ . This loss is computed at  $u$ .

So, this is just saying that if I am going to apply throughout all the rounds  $u$  always this is the loss I am going to get. I am going to compare this loss with whatever the loss you are going to incur by your algorithm. So, this is the loss you are going to incur right. So,  $w_t$ 's are the one selected by your algorithm. We are going to argue that this guy is upper bounded by this quantity which is at every point at every  $t$  we are basically looking at  $f_t(w) - f_t(w_{t-1})$ .

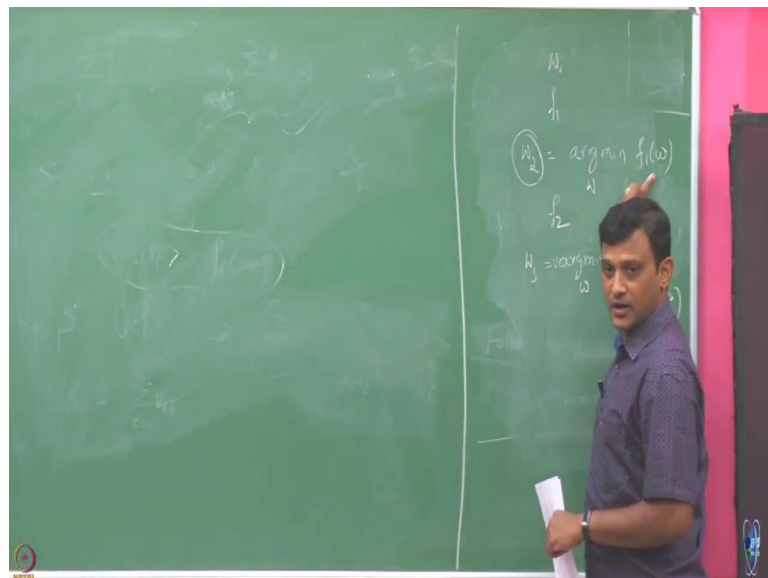
Student: (Refer Time: 03:09) next.

It is the next. So, if  $w_t$  is the one selected in round  $t$ ,  $w_{t+1}$  is selected in the next round ok. So, we are basically represented everything in terms of the difference between  $f_t$  function computed at  $w_t$  and  $w_{t+1}$ . So, this is basically this is basically the major step in this, after that it will be easier for us to show depending on what function  $f_t$  is the regret bound on this. We are going to show this ok.

So, we want to show this, showing this is same as showing like if see the first term is going to be the same in both of these. So, showing this is same as I want to show that this regret is this is going to be showing same as I want to show that  $t$  equals to. So, if I am going to look into this, if I just knock off this term because the first term is same; I want to show that this guy is upper bound, this is a lower bound on this. So, I am just rearranging these two things.

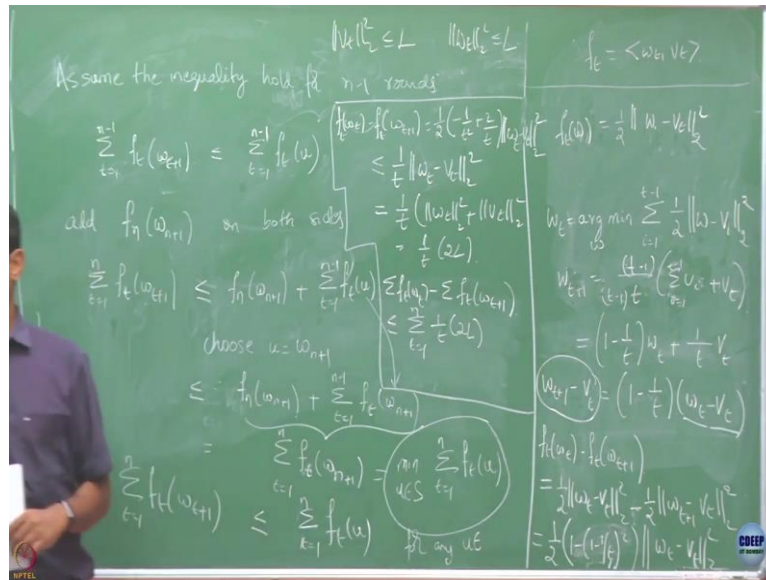
So, is it fine, like I want to show this is a upper bound on this; I am just saying that is same as saying that I want to show that this guy. So, if you just like rearranging these things ok. So, we are going to prove this by induction. So,  $t$  is running from 1 to  $n$  right. So, let us take  $n$  equals to 1 first round; is this true for first round? Just by this argument right.

(Refer Slide Time: 05:18)



So, in the first round as soon as you get  $f_1$  we are going to choose  $w_2$  which is going to minimize this function  $f_1$ . So, for  $n$  equals to 1 you have  $f_1$  of  $w_2$  which is definitely going to be smaller than any point  $u$ . This is by that way you are choosing  $w_2$  ok.

(Refer Slide Time: 05:53)



Next we are going to select assume that this inequality holds for  $n-1$  rounds. And, then we will show that it will also holds for the  $n$ th round ok. So, then it is like  $t$  equals to 1 to  $n-1$   $f_t(w_{t+1})$  less than or equals to equals to  $n$   $f_t(u)$ . So, notice that this  $u$  is any number in my  $S$ , but fixed right from the beginning. Only this  $w_t$ 's are varying as per your algorithm.

Now, what we will do is I am going to add on both sides ok, I am just going to add this both term. So, if I add this both terms, this is going to be  $t$  equals to 1 to  $n$   $f_t(w_{t+1})$  like I add this and absorb it inside the summation. And, this is going to be upper bounded by ok, I have just added. So, this instead of  $n-1$ , I have written it  $n$  because this term is.

Student: (Refer Time: 07:48).

This one.

Student: Above (Refer Time: 07:53).

This one.

Student: Yes.

What you are saying this should be.

Student:  $t$  equals to 1.

So, as I said this is this  $u$  is arbitrary right like this  $u$  is any  $u$ . So, this particular  $u$  I have chosen, this holds for any  $u$ . So, I am going to choose this  $u$  to be  $w_{n+1}$ , that specific value chosen by that algorithm in the  $n+1$ -th round. So, if you do that what is this going to be?  $f_n(w_{n+1})$  plus 1 plus  $t$  equals to 1 to  $n-1$   $f_t(w_{n+1})$  right.

I am all I am doing this doing this, this should hold because this is true for any  $u$  and so far particular  $u$  equals  $2n$  plus 1. But, now if you look into this, this is nothing, but they will add up, but I do not want that. I want them to add up on  $w_n$  plus 1, I want them to add up because I know how  $w_1$  plus 1 is selected. They are selected in a specific fashion right, but you will just a fixed quantity for me given thing. So, I can still add up right with  $w_{n+1}$ .

But what is this by definition? This  $w_{n+1}$  is such that this is basically min over right, because  $w_n$  plus was selected basically by minimizing this over all  $u$ ; that is why this inequality this equality holds here. So, what we have done? We have shown that this is upper bounded by the minimum of this is with are we done with this. So, this is the minimum over all  $u$  right.

So, if you take a one particular  $u$  from this, it is going to dominate this. So, I can now take that like this is true; this is true if you whatever that initial  $u$  you started with you just take it; it has to dominate this. Now, this is like any  $u$ , this is for this is for any  $u$ . So, recall that the way I am doing I am first saying that this inequality holds for any  $u$  belong to  $S$  and now I am trying to do induction.

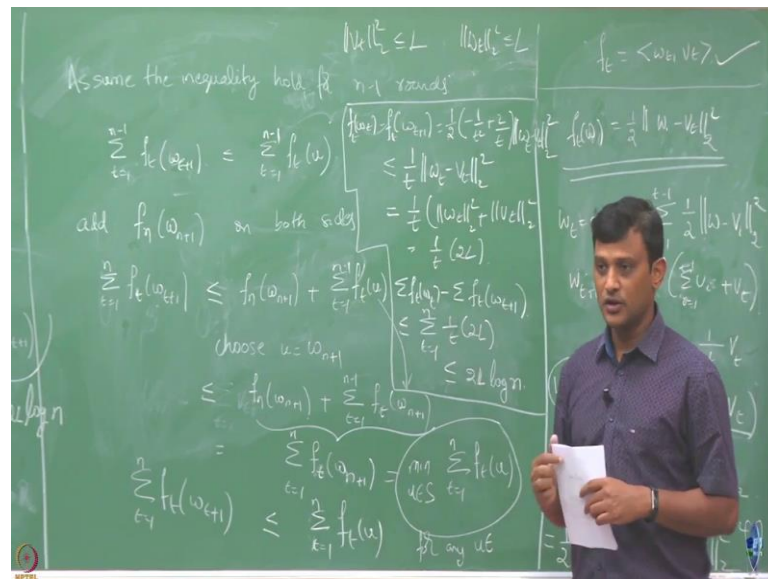
When I came to this step, since this is true so far for any  $u$ , I took a specific  $u$  to be  $w_n$  plus 1 and then I got on this ok. But, now this is the minimum over all  $u$ ; so, this should be if you take a any particular  $u$  this should be an upper bound. So, what I have shown is now exactly what we wanted. So, then all we need to prove any regret bounds all we need to worry about is this upper bound.

If we can give an upper bound on this quantity, we have an upper bound and also the regret as if we can show an upper bound. See now this guy is no more specific on  $u$ .

Now, if I can bound this quantity, this will be a regret bound on this and that is holds for any every all  $u$ . So, that is what my regret definition was ok. So, all of you followed the steps here ok.

So, only trick we used here is after adding this we made sure that this guy is basically by choosing this  $u$  in a sub in a specific fashion, this becomes an actually this is this  $w_{n+1}$  is coming by choosing this minimization problem like this, by all the previous in  $f_t$  functions fine. Good, we have this bound expressed in this form now; let us compute this what happens for a specific convex function.

(Refer Slide Time: 13:35)



Earlier one possible convex function we have done is like some  $\langle w_t, v_t \rangle$ , but I will not directly jump to this; instead we will look into a quadratic loss function. So, we are going to take  $f_t$  which is defined as; so, all of you understand what is this? This is a squared  $L_2$  norm and I have included this factor half there.

So now, let us try to compute what happens when my functions are  $f_t$  's like this ok. So, in each round what I am going to do? I am going to compute my  $w_t$  to be arg min of my  $\sum_{i=1}^t f_i(w)$  equals to  $\sum_{i=1}^t \frac{1}{2} \|w - v_i\|_2^2$  this quantity right half of  $w$  minus  $v_t$  equal  $w$ . Can somebody compute this quickly and tell me what is  $w_t$ ? This should be easy to compute right, you can differentiate this.

Student: (Refer Time: 15:04)  $w_1$  choose.

No, how were we we wrote right a scale algorithm. So, how did it choose?

Student:  $w_1$ .

$w_1$  ok, you are talking about  $w_0$ ; you choose in any way you like that only we are only worried about subsequent ones. So, this is so, and  $t$  equals to 1; I do not have anything to optimize here, I will choose  $w_1$  arbitrarily, but after  $w_t$  equals to 2 I am going to start optimizing.

Student: Mean.

Which is going to be just the mean of all.

Student:  $v_t$ .

$v_t$ 's right. So,  $w_t$  is going to be; so, it should be  $v_i$  here now ok. So, this is for  $w_t$ , if I am interested in  $w_{t+1}$  it is just going to be like this. We are going to basically average first  $t$  terms for the  $w_{t+1}$  th prediction, now this can be kind of just re-written. So, you can go and verify that, you can rewrite it in terms of this. This is easy to observe like you can split the summation over  $t-1$  terms.

And, the average of a first  $t$  minus terms is going to be  $w_t$  and then the last term comes like this, you can just verify this is just like manipulation. So, if you just add  $t-1$   $t$  minus 1 and then split this as. So, I will just end up with this ok. Now, I want to further simplify this.

So, if I am going to simplify this as minus  $v_t$ , this term is going to be  $1$  minus  $1$  by  $t$  into  $w_t$  minus  $v_t$  ok. So, basically what I did to come from here from  $w_{t+1}$  I will just subtract minus  $v_t$ , we also subtract minus  $v_t$  from here. And, you will have this  $1$  by  $1$  minus  $t$  common term and we will end up with this term.

So now, go back and plug this relation in what we are really interested here, I am going to take one specific term here  $f_t(w_t)$  minus  $f_t(w_{t+1})$  here. Now by definition what is  $f_t(w_t)$ ? It is  $f_t$  computed at  $w_t$  right this is nothing, but already  $f_t(w)$  which is in this fashion. So,  $f_t(w_t)$  is half of and the other quantity I am going to get.

So now, what I will do is I have already expressed  $w_t$  plus  $1$  minus  $v_t$  in terms of  $w_t$  minus  $v_t$ . I am going to plug back that relation here from here this quantity and if I do

this I will end up half of  $1 - \frac{1}{t^2}$  whole square times. So, is this correct? Like I have just substituted this by this quantity and then I have just pulled out the common term  $v_t w_t - v_t$  whole square.

Now, if you just expand this square term here, this is going to be  $1 + \frac{1}{t^2} - 2 \frac{1}{t}$  and then if you knock up the  $1$  term here, what you will end up is  $f_t(w_t) - f_t(w_{t+1})$  is equals to half of  $1 - \frac{1}{t^2}$  by  $2$  by  $t$  times the norm right. This is we will be this is squared right. So, this we will going to get  $1 - \frac{1}{t^2}$  square.

Student: (Refer Time: 21:21).

So, we have substituted this entire thing here right. sorry there is a minus here right which I made a mistake here, this mistake this minus I have to write here not it was not plus, is that correct now ok. Now, what I will do is I will ignore this  $1 - \frac{1}{t^2}$  term because that is having a negative sign I will ignore that, then I am going to get an upper bound. And, that is going to and this  $2$  knock off with this, I am going to get  $1 - \frac{1}{t}$  times  $1 - \frac{1}{t^2}$  ok.

So, at this point let me make an assumption that this loss vector that has been chosen here. So, here notice that the environment is choosing the loss vector  $v_t$  and I am defining the loss like this. So, suppose assume that my loss vectors are all bounded by some number  $n$ , some number which I know and it just a bound.

So, if you have this you know that my  $w_t$  here is basically finding average of my  $v_t$ 's right;  $w_{t+1}$  earlier we wrote this is nothing, but the average of this  $v_t$ . So, because of that is it true that my  $w_t$ 's will also satisfy this condition? If all my  $v_t$ 's are upper bounded by this because  $w_t$ 's are nothing, but just average of since  $v_t$ 's; they should also be satisfying this condition.

Student: (Refer Time: 23:29).

We can assume that. Let us see do I need to do this, this is a starting from  $t$  equals to ok; the only issue that comes is only time when  $w_1$  is coming is this point when  $t$  equals to  $1$ , otherwise it is coming nowhere. So, that is you can always split this part for  $t$  equals to  $1$  and take the other part ok. So, you have separated that out.

So, let us not; so, let us assume that  $t$  equals to 1 part we have ignored this rest only focus on the other term now, but for all the other terms you agree that this holds ok. Now, issue of now you have these now you go and apply triangular inequality on this. If you go and apply triangular inequality on this, we are going to get this is upper bound as norm of  $w_t$  square plus norm of  $v_t$  square, that is one bound I can get.

And, I know that both of them are upper bounded by  $L$  so, I will end up with 1 by  $t$  times  $2L$ . Now, what I will do is I will take the summation on both sides that is what finally, what I am interested in this right. So, this is going to be what then? Summation  $t$  equals to 1 to  $n$   $(1/t) 2L$ . So, what is summation 1 by  $t$  as  $t$  goes to 1 from  $n$ ? What is the bound I can get on this?

Student: (Refer Time: 25:37).

Log  $n$ . Why is that? Because, integration of 1 by  $x$  is  $\log x$  right and here we are doing discrete, but fine I can make it continuous and get an upper bound. So, this is going to give me  $2L \log n$ . So now, what I have is a bound which is independent of what was the  $u$  I am interested in, this is true for any  $u$ . So, finally, what I ended up showing is for the FTL algorithm this is upper bounded by  $2L \log n$ .

So,  $L$  is some constant which I am assuming that is the bound on the norm of this  $w_t$ 's. So now, if you see that finally, what I am able to show that the regret of my follow the leader algorithm is upper bounded by. So, is this sub linear this regret? Yes, right. So, if you are going to.

Student: (Refer Time: 26:56).

So, we have shown that this follow the leader algorithm has this good regret bound which is  $O(\log n)$ . Then the next question is fine if you have taken a quadratic function like this you are able to get it, but is it true for any convex function? Or is it true that if it is true for any convex function why did not you consider this function? And, show that this also holds you and if you have function like this then also this bound holds ok.

So, we will argue it in the next class that if you are going to just apply follow the leader algorithm in this fashion, when we have this linear functions like this; we may not end up with a bound like this. In fact, we may end up with a linear bound ok. So, something



more has to be done there. So, for that we will see that we have to kind of do a regularization on this. It is not that we should arbitrarily allow this  $w_t$ 's to change too much, should not possibly allow them to change them too much.

We should bit slightly constraint them. Then we will show that if you can do that then even for this linear case we will end up achieving a could bound like this. And, then we will show that any convex function which need not be just like linear like this, that can be translated to be a linear function. And, then the regularized version of the follow the leader algorithm will you also good performance there ok.

So, let us do all this in the next class.