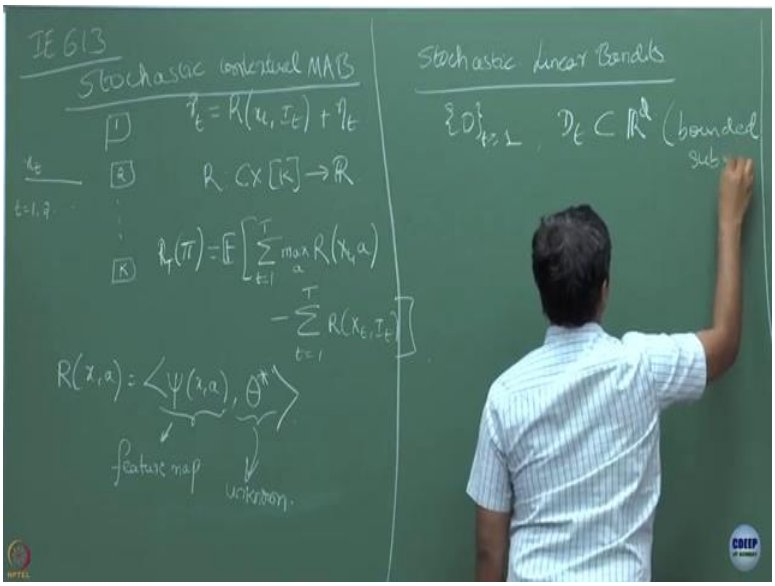
Bandit Algorithm (Online Machine Learning)
Prof. Manjesh Hanawal
Industrial Engineering and Operations Research
Indian Institute of Technology, Bombay

Lecture - 44
Stochastic Linear Bandits

So, let us continue our discussion on multi-arm contextual bandits that we started.

(Refer Slide Time: 00:32)



So, just a quick recap on what the things we did. We say that we have in every round a context is revealed to us and looking at this the learner has to figure out which action to play and we said that we have about k actions, ok.

Then we said that when he is going to play an action, I here he is going to get a reward which we said in round t is how did we write the reward. We said reward in round t is some function which depends on the context in that round and the arm he played plus a noise term. And for this, we wanted to see how we should be choosing an arm in every round looking at the context. So, that my regret is minimized right.

So, how did we define the regret? The regret was, so the regret of a policy pi we defined it to be expected value of. So, this is the context X_t, then we said. So, I_t is the arm played by the learner using policy pi in round t. So, this is the cumulative I reward he would have got and this is the best you could have got in every round, ok.

If you knew this reward function.

Student: (Refer Time: 02:50).

So, whatever like in X_t when he observes what is the best he could have gotten in that round t, this is the reward he is going to get in round t and this is the sum over total rounds t and this we are comparing with what would I have got gotten if he has played I_t in round t.

So, then we said that, so what then we said that yes as of now this reward function is just a function of two variable, that is the context in that round t and the action you play in that or like it is just a like reward is a function from your context set and action set to some number reward function. The learner does not know this, ok. He has to figure out.

So, then we said that if there are only let us say finitely many contexts, then what learner can think is for each context and the arm pair he can think that has an another arm whose rewards he do not know. So, in this case instead of thinking as k arms, he can think of for each context and an each arm he can think it as a pair and across this pairs he can think them of individual arms and then try to learn the mean value of that.

Then this is corresponding to the standard k arm bandits. In that we have, but the k corresponds to the number of context into the number of arms there.

Student: (Refer Time: 04:42).

And eta t we just said this is what did we say? It is a sub Gaussian noise.

Student: It has zero mean.

It has a zero mean noise, but we said that if that is the case like if the number of context are huge, then I am basically learning over a large number of arms which if I am going to just apply the standard bandit algorithms, its regret is going to be like scaling like how many pairs are there, right.

Then we said that say whenever there has a contextual case, it is not that rewards across these arms are independent. May be reward from one arm for a context is somewhat similar to that you can observe from other.

There could be a potential correlation that is like if I have a context like this, if I am going to observe some reward for this context from these arms and if I get an another context and if I observe rewards for that new context across these arms, potentially there could be some correlation across them, right because if the two context let us say happens to correspond to same like n user who is trying to log in maybe their interest could be potentially similar. Because of that I can expect as kind of similar reward across these arms.

We are going to henceforth assume a structure on the reward function which is of this form where this phi function what is what would we call this phi function?

Student: Phi function.

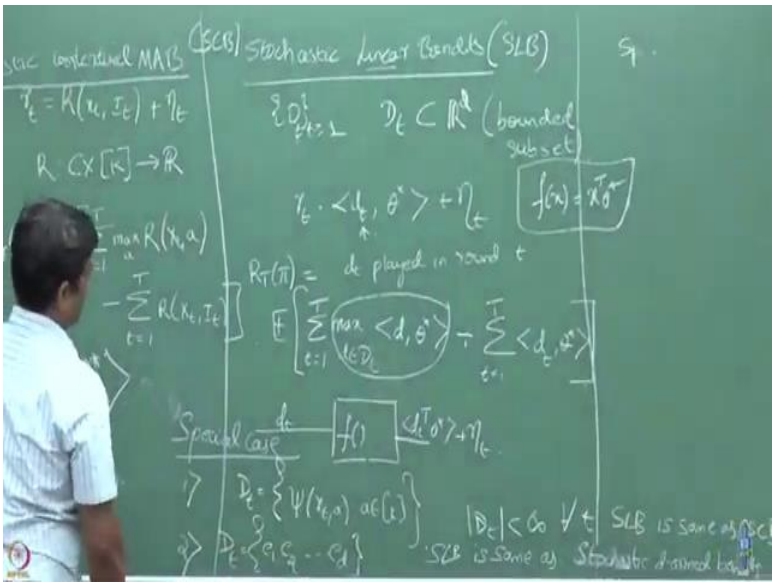
We called it as feature map and this function is known, this is known for every possible pair of context and arm and then we said that this is an unknown parameter which is independent of context arm. It is just theta star which is going to parameterize this reward function.

So, this is one setting we have, this is what we called as, so we called it stochastic contextual multi-arm bandit or contextual stochastic multi-arm bandit?

Student: Stochastic sorry.

Stochastic contextual multi-arm bandit.

(Refer Slide Time: 07:51)



Then we said that we have another setting called stochastic linear bandits. So, let us now treat this as like a different problem for time being and then let us try to connect these two. So, what we said is in every round a decision set is revealed to you.

So, these are the sequence of decision sets and let us say D_t is some subset of some R^d, but let us assume these are all, this is a bounded subset. So, it is not I am not saying that D_t has finitely many elements in that let D_t could have uncountably many, but the thing is each vector here is bounded.

That means, each component in the vector is bounded. Now my goal is to in every round. So, if you are going to pick, so the, so this is D_t here, here the reward you are going to obtain in round t is given by what is that it is already linear.

So, I am already looking into the linear bandit. This is going to be given by D_t into let us say theta star plus noise. So, what is D_t? D_t is the arm plate in round t. So, now I am talking about arms are nothing, but the set of feature vectors. So, D_t is a set of feature vector that is revealed to you in round t and what you have to do is, you have to select a feature. If you select a feature D_t here, this is the reward you are going to get.

Now, what is the maximum expected reward you are going to get in a round? The maximum reward you are going to get in a expect the best thing you can get in a round is, so over a time period T if you know theta star already, the D from this decision set

you have chosen is the D that maximizes this, but if you are going to play whatever your algorithm tells you is let us say D. I am going to make D theta star and let me call this as R_T of pi. So, this is what we are going to call it as stochastic linear bandit setting.

So, what is I am doing there, I am basically trying to optimize a linear function here, ok. So, let me put it in other way. I have a function f whose output is x star theta. So, let us say I have a function f which is linear if I know. So, this linear function is parametrized by what theta star if k know x, I know what is the output of this function.

Let us say you have a black box which is this function. You do not know what is this function, but you want to find an x that maximizes this function f, ok. So, how you are going to do this every time? You are going to choose a set D_t.

What is going to happen is, you do not observe theta star here, but what you observe is the output of this function. What is that is yet if we are going to give it, if I give D_t as the input for this function, its output is, but the thing is you do not observe it. You are going to observe a noisy version of this.

So, that is what I say that is the reward for you, but now your goal is to find a point from the from decision set every time such that you are going to get the highest reward, ok. So every time you are going to play, you are going to observe this. It will this output is going to depend on theta star. What you are going to just observe is this inner product plus noise added to that, ok.

Now, your goal is to in every round you want to match this. This is the best you would have gotten right if you know theta star, but you do not know theta star. What you are getting is only these observations, noisy observation from this you want to. So, this is minus. Here you want to learn this theta star, such that every time you are going to play an x which optimizes this function.

Well, then this is exactly this is stochastic linear bandit or this is you are trying to optimize a linear function. Are you following? I have just a linear function here. I do not have directly access to its parameter, but what I can do is in every round I can play a point, I can query it.

So, this is like a query box, a black box to me. I can query with whatever input I want from that. I am going to observe this up to output. My goal is to get this output which is as maximum as possible. So, this is the best I could have got, but if I am using going to choose d t in round t, this is what I would have gotten.

Student: This represent the error made in features case.

Whether this noise represent the error we made in features?

Student: Features case.

Yes.

Student: It is that some kind of model.

At least in this model right, we are going to treat it as some observation noise. It could be like anything like you remember we assumed already what it is we said that this is a conditional sub Gaussian noise, right. So, that is given your observation till time t. This noise could depend on all the things you have observed so far and, but it is a condition on that it is a sub Gaussian noise.

Student: Can directly signify and let me show the rewards?

So, you are coming to this right like what is the guarantee that there is some reward function, but what I have done is basically parameterize through these features. If I am going to have a features like that, is it guaranteed that whatever that function is, it is a good approximation of that function. Is that your question?

Student: Right.

I mean I do not think that eta corresponds to that. You are kind of assuming that. Yes, there exists a good features map such that this is true. We are not saying that by there exists some noise. I can add a some noise correction term here, such that whatever this function is that is exactly represented to this linearization part we are not saying that.

We are just saying that this holds, but whatever eta corresponds to here is the observation noise. I am making whenever I am paying a particular feature, ok. I have written them separately because this is like stochastic contextual problem, this is a linear optimization

problem, stochastic linear optimization problem where I have this function which I am trying to optimize over many rounds. So, this is what I have collected from it, but this is what I would have ideally like to collect from it. I would have collected this had I known this theta star, but I do not know this.

Now, in the last class we just said that this problem and these problems are identical, right. How did we say that? What are the how did we map this problem to this problem or this problem into this? So, we just said that D_t could be.

So, D_t here is nothing, but the feature maps for the context and all possible actions we have, so that I could treat it as D_t that is revealed to me. So, it is true that right as soon as I see the context xt here, I have assumed that I know this phi function. I could compute all of this. I am going to take that as my D_t here.

Now, the question is from this D_t I am trying to look for a feature map that is maximizing this, then it is same as asking for a arm which maximizes the reward for that context, right. So, that is why we said that. So, now everything remains there once. I do this mapping and assume that these D_t set is finite for all t because I have only finitely many arms here right k. There are only k arms here or k actions here if once I assume that this D_t is this. So anyway if k is finite, this set is anyways is going to be finite.

So, then this actually take considers this problem as a special case when I took this stochastic linear bandit, right. I said D_t to be a bounded set, but it could have uncountably many.

But suppose now I say that I make this D_t to be finite. If I can take this, solve this problem for uncountably many, I should be able to do this for finitely many also, right. So, as a special case.

This is just a cardinality of size set D_t. So, if I as a special case in this if I am going to take this D_t to be this feature vectors for each arm, in that case D_t is finite and then apply this problem and these are same.

Student: So here experience want to fix that capital C and x_t.

So, that is what yes x_t 's are coming from some context at C. Whenever I got this x_t, I went and mapped them to this features, ok. This could now lie in a dimension in a space of dimension same as theta star.

Student: Which is small d.

Which is small d. Now I am just saying that d is that that set of feature vectors itself that is my decision set in the stochastic bandits. I am just the what we have doing through this is there is no c here. It is just like a decision set that has been coming to that have been given to you in every round. From that you have to identify which is that feature which is maximizing this function.

Only thing that is coming for you is D_t set here and now, we are saying that yes that the identity of the marks arms here is not important because I have mapped them to this feature vectors. What matters to me is now this feature vectors and now once I have a mapped this arms to this feature vectors and now it is just about thinking which is this feature that is going to maximize my reward.

So, that is why I can think this problem. I just like a is just like a stochastic linear bandit problem or just like a linear stochastic linear optimization, ok. So, now we have done this abstraction. Now we can just try to focus on this problem rather than just assuming D_t is finite in every round.

We will allow D_t to be in anything, in each set. We just want is it is coming from a bounded set in this setting, I am find D_t to be a bounded subset of R^d. It could be anything. If I have it is only consist of only finitely many points in it. It is still a bounded set of R^d.

Student: This we say let us say for solving this linear problem.

Solving I do not care whether on this set whether I am finding whatever like. See I am only finding point from the given set D_t. From that whether I am able to find the best point, whether it is it is consist of uncountably many or finite, I do not care.

I have to, all I have been given is this D_t. From that I want to find a point which maximizes this.

Student: Right.

What you have used optimize it, I do not care that is of irrelevant to me. All I care is what is the optimal value you use gradient descent or some optimization, I do not care. So, right now we are not talking about how we optimize, right. We are just talking about. This is what you want whatever this value is, how far you are by using your policy compared to this. That is the only thing I am worrying about.

So, as long as this D_t is a bounded subset of this R^d, I am fine whether it consists of finitely point or uncountable, I do not care. It all falls in this setup, fine. So, now the question is how to go about solving this. So, if you recall in our last class, at the end we also discussed that if I allow my this D_t to be unit vectors, then I can think of this problem is actually solving a stochastic k arm standard multi-arm bandit problem, right with theta star equivalent to the means of the arms, fine.

So, let me just write it special case. So, these are actually special cases. So, special case. So, if I let this case, then my let me call this as SLB and let me call this as SCB Stochastic Contextual Bandits.

So, then it is like SLB is same as SCB right. If my D_t is happens to be this set in every round, then it is nothing, but stochastic linear bandit is nothing, but stochastic context. The second case if my D_t is e_1, e_2 all the way up to e_d ok, so where D is the dimension of this theta. So, then what is this problem, what is then stochastic linear bandit is same as with D arms.

Student: D arms.

Stochastic multi-arm bandit with stochastic D arm bandit, ok. So, that is fine. So, now we have bit. So, we have bit now both this contextual bandits and this linear bandits, we have specifically studied them earlier, but now we are going to study a bit more general structure which is now a stochastic linear bandits.

So, just for this I am going to just write this as without expectation as the regret of a policy pi whenever I want to find expectation. I will write expectation on this directly expected value of R_T, ok. How to how we go and solve this? So, what is the unknown in

this theta star if you can somehow figure out, what is the theta star, then we are more or less than right?

So, in the multi-arm bandits k with k arms there mu 1 mu 2 up to mu k, they were unknowns. If you figure out that mu 1 mu 2 mu k that problem is I can solve it. Now here theta star is unknown, but theta star is not like one real number, it is a vector. If I can figure out this theta star, I can solve this problem how to figure out this and how to go about it.

Student: Right.

So, I see the D arm bandits that means k is already D, you are treating with k d arms in that.