Transcriber Name: Saji Paul
Statistical Learning for Reliability Analysis
Prof. Monalisa Sarma
Subir Chowdhury School of Quality and Reliability
Indian Institute of Technology, Kharagpur

Lecture - 56
SVM (Part - Ⅲ)

(refer time: 00:31)

Hi guys, so in continuation of our lecture on SVM, so this is the third lecture. So, in our last lecture we have seen lagrangian multiplayer method how lagrangian multiply method we can use to find out an optimization problem. So, this will be using now to solve the maximum margin hyper plane. Then finally we will see multi-class classification using linear SVM. Till now we were just discussing about the binary classification.

Where there is two classes, we have taken one is plus one is minus this is just an entry of use. So, now we will also see how we can do multi-class classification using linear SVM having said that as SVM actually it works best when we work with binary classifications.
(refer time: 01:11)

So now linear SVM. So, basically, we need to solve the maximum margin hyper plane to solve the linear SVM. So, what was our now we will we have learned lagrangian method how will solve lagrangian multiply methods. So, what will be done now lagrangian will have to now write the lagrangian. So, our constraint was; remember our constraint >= 1 this is our constraint remember this was our constraint.

Our problem was minimize this subject to this constraint. So, this was our problem we have formulated our problem in the last lecture. Now in the lagrangian inequality constraint we have seen that was not >= 1 but it is <= 0. So, when it is the same just the inequality constant here, we have used <= 0 here since β =1, so what it will be we will just change the sign so I can form this is this will be my lagrangian.

Instead of minus instead of plus here I will use minus because of this constraint. Since my constraint was greater so I have changed the sign to make it less so this is my lagrangian now. So, there is my only but so this is my lagrangian now I will have to find out the KKT constraint.

So, where λ i’s are lagrangian multipliers, we have already seen W and b are the model parameters.

Now one more thing to be noted here so, my minimize the function was this was the function here I have used square W ^{2}. So, we use this instead of this for the sake of simplification in conclusion and it does not alter the programs called programs goal adversely. So, even if we use W ^{2} it does not affect the program call it remains the same our value will not be affected. So, you will see if I use square my simplification becomes easier.

So, I have used instead of W / 2 that is the magnitude of W / 2 I have used the magnitude square by 2.
(refer time: 03:16)

So, now what will mean by my KKT constraint? Remember the KKT constraint the lagrangian first order differentiation of lagrangian with respect to all the attributes. So, whatever one is W, one is b that I have done and then equalize it to zero of course then del L / del W I got this so, from this expression I got W = ∑ λ i y i x i. Then when I do del L / del b this is what I got this expression you can see.

If I do del L / del b what I get this is only from here I will be getting is not it. So, I will be getting this from λ i y i and then what are my other three constraints in KKT constraints there are other three constant that is λ I should be >= 0, this is one constraint. Another is λ into x i = λ into h i that means the constraint so this is λ into this is my h i this is equals to 0.

This is another constraint third constraint is my all my h i's, h i should be <= 0. So, this I can write it into less equals to format, this is the constraint basically at a constraint whatever format you write. So, these are my three KKT constraint along with this derivation. So, now I will have to solve this to find out the value of the W b and x and y. So, solving KKT constraints are computationally intensive.

So, it is like the example what we have seen that was very easy so we could easily solve it. But then for this it is not very easy to solve it directly by hand you will have to use one of the numerical technique methods I am these numerical techniques I have mentioned in many places. Of course, now nowadays like you do not have to know a numerical techniques the softwares are available you just have to give the data and you will get the solution.

But I my suggestion is that of course let the computer do it let we have the ready-made code and we will run the code, fine I am not doing it but I still I should still know it is not it, you should know the walking of the matter. So, please this numerical technique methods not only for discourse for many things you will be needing in many of your studies you will be needing it. So, please learn in different numerical techniques even you have not learned it till even if you have not learned it till now.

So, using any of the numerical techniques you see which numerical technique will fit and then we will solve the we will get the values of this.
(refer time: 05:44)

So, we first solve the above setup to find all the feasible solution, then like we have seen we will find all the physical solution feasible solution based on the different constraint. And then once we get all the feasible solution then we will among the physical solution which will give us the minimum velocity because there is a minimization problem is not it. So, which will give us the; minimum value that is my optimum value.

Then we can determine the optimum value for this, optimum value for the different parameters. So, lagrangian multiply λ i must be 0 so here we have got this λ i h i = 0 is not it. So, when h i = 0 λ h i = 0 either λ is 0 or h i = 2 when h i = 0 that means my when my constraint is equals to 0 when what will my constraint will is equals to 0 what when that will happen, when any points that lies on the line then it will be equal to 0 is not it.

Any points that lie above the line it will be positive any points that line below the line will be negative, so any point that line on the line it will be 0. So, as I told you the point that lines on the parallel what to say the parallel lines of the hyper plane that we call it the support vectors. So, here that is written lagrangian multiplier λ I must be 0 unless the training is an x i satisfies the equation this.

If it satisfies the equation if it satisfies this equation that means h i = 0 is not it - 1 = theta I can write this y i W dot x i + b - 1 = 0 if this is satisfied then λ is not equals to 0, if this is satisfied then what happens then λ i is 0. So, here it is written lagrangian multiply λ i must be 0 unless the training instance satisfies this and when the training instance satisfies this one when it happens.

When the training instant will satisfy this equation when the training instance are support vectors. When it is support vectors then only it will fall in the line it will fall in the line that means then we will get this is equals to 0 is not it. Does the training tuples with λ i greater than 0 lie on the hyper plane margin enhancer support vectors. When this is 0 λ i’s not equals to 0.

So, λ i’s λ i value already we have seen λ i greater or equals to 0 either equals to 0 greater than 0 when λ is not equals to 0 means λ when λ is not equals to 0 my h i = 0 when will by my h i will equal 0 under my points lie on the line on the hyper plane is not it. So, points that lie on the hyper plane what is the support vectors. So, with λ i greater than 0 are support vectors.

The training instance that do not lie on the hyper plane margin half λ is equals to 0 that is why I remember when I told you this support vectors give and last like the support vector gives us lots of informations. Support vectors gives us the information when λ i this is with λ i when λ i is greater than 0 then for all those instance λ i is greater than 0 those are support vectors.
(refer time: 09:11)

So, for a given training data using SVM principle we obtain a maximum margin hyper plane in the form of W b and λ i's is not it. We form a what is that all those KKT constraint and all this is nothing but our machine is not it that is nothing but our black box. Basically, from whim that is our function let me tell you that is our function, that is a function given any output it will use that function inside a black book.

And we get the output raise output it files in this class or that class, remember we are discussing binary classification two classes. So, this is called this black box this function we call it SVM support vector machine. Now let us see how this MMH can be used to classify test tuple say X given any test tuple X given any observation X how we will classify whether it falls in plus plus or it falls in plus minus.

So, we will find out this delta of X basically this is the representation we find the delta of X. So, it gives a magnitude and S sign and sine if it is lie above the hyper plane then it is under positive class if it is like below the hyper plane it is a negative class. And magnitude is how far it is from the hyper plane, lesser magnitude is lesser nearer to the hyper plane more the magnitude is far down further from the hyper plane.

So, how we can find? So, λ of X we can write as w dot X + b this is the dot product mind data W dot X + b and what is W we already found out from this lagrangian multiplier this is my W so dot X + b so this is W I have found out from the lagrangian multiplier.
(refer time: 10:54)

So, this is famously known as representer theorem this is known as representer theorem which state does a solution W always is represent as a linear combination of the training data. The solution W is always represent a linear combination of the training data you see linear

combination is not it, y i is the class x i is the input tuple it is the input basically and λ i we already know. So, W is a combination of this is a linear combination of this is called the representer theorem.
(refer time: 11:39)

So, the above involves a dot product of x i dot X and this say when you want to find out the class of X so I use and this is a dot product you see. Earlier there was a dot product of W dot X that we have from brought down to there is a dot product of W i dot X so x what is x i? x i all are all the initial observation that is we can say is the training observation the trading training data. So, we will have to find out the dot product of x i dot x and x i but this is from i = 1 to n, so this is a dot product.

So, in this above involves a dot product of x i dot X where x i is a support vector this is so because see here where the x i is a support vector why, if when λ i is not equals to 0 then all the for all those tuple λ is not equals to 0 those are support vectors in it. If λ i is equals to 0 then this equation will be λ is equal to 0 and this one becomes 0 only is not it. So, when I have X of when this is when I have this that means my λ i is not equals to zero.

For all those λ is not equals to 0, my x i are the support vectors. So, where x i is a support vector we now we find out the sign of λ of X sign as well as the magnitude of the λ of X. If it is positive then x falls above the MMH is and so the SVM predictor class belongs to class level plus if it is if the sign is negative then X falls on or below MMH and the class prediction is minus.

Now once the SVM is trained with training data the complexity of the classifier is characterized by number of support vectors. So, once we have trained a model, we have trainer model means basically we found the function, I found a function by which we call from which we found out the different parameters. Once we have done all those now the when we are into class given any tests tuple given any observation, we have to find out it belongs to each class.

Then what is the correct what is the complexity of that is this basically is not it; this calculation will give us the complexity is not it. And what is this how it is totally determined by how many number of support vectors are there because only for those support vectors will be getting this expression. If these are not support vectors then λ i will be equal to 0, then this expression will not be there.

So, we will have to do this calculation only for those X which are support vector. So, complexity is given by the support vector that is what I remember last lecture there was one point I told will come later discount support vector gives lots of information the support vector gives the complexity of the classifier. So, the complexity of the classifier is characterized by the number of support vectors clear this is.

So, dimensional data is not an issue in SVM unlike in other classifiers, in (14:49) you have seen dimensionalities that we have faced the cards of dimensionality problem is not it. When we try to find out the distance heater if it is dimension is quite high then computation it becomes computational very intensive. But in SVM dimensionality is not an issue we will be seeing that in one of the lecture later.

But now for you to remember this dimensionality of data is not an issue in SVM. Basically, in SVM what we are doing we are just trying we are just finding a dot product this dot product. So, when you are finding the dot product here even if it is if it has a very high dimensional it will not have an a computational will not be not too much.
(refer time: 15:28)

So, now let us see an example. So, consider the case of a binary classification starting with a training date of eight tuples. Suppose these are the training data and it has two attributes A 1 and A 2. Now these values are given λ is values are not given suppose these values are given and for each values this is the training. So, training means the class level is mentioned for which we are getting plus class for which we are getting minus class it is given.

So, and then we will be using lagrangian multiplier using lagrangian multiplied and we will we will solve the values of the we will find out the different values. So, first we found out the λ values. So, using we can solve the KKT constant to obtain the lagrangian multiplier. So, if we found out the λ i for each tuple when we calculated we found a λ i for this double is this then right for this tuple is this for other λ i we found it 0.

That means what from this what you can conclude that means this two are these two tuples are the support vectors. These two all these values these values this value this value and this value it falls above the hyper plane and this and this it falls below the hyper plane that it belongs to minus plus below the hyper plane this plus it belongs to last class it falls above the hyper plane. And this two one belongs to plus and one belongs to my minus these are the support vectors that is why value of λ i is not equals to 0.
(refer time: 16:55)

Now what happens, once we know once we know the λ i, we this given the X and the x 1 and x 2 are given. So, what will be will be able to find out that w 1 w 2 what we have to find out we have to find out w 1 w 2 basically we have to find out W total what we need to find out we need to find out λ we need to find a W, we need to find out the b is not it. So, we found out λ now we will be able to find out w 1.

So, what is w 1? w 1 we have already seen here this is λ w 1 W is not it. So, this is W, so W means since there are two parameters so there will have w 1 and w 2 dimensional. So, that what will be w 1? w 1 will be this λ i y i x i x i 1 w 1 will be this parameter for the to w m parameter will be taking this w 1 x 1 is not and w 1 x 1 w 2 x 2 for w 2, I will be considering this for w 1 I will be considering this.

So, see here λ 1 65.52 and y i class is plus so I am taking is one for plus class I told you already I am taking 1 for minus plus I am taking minus 1. So, then y of i 1 y of i 1 is 0.38 then again, I will be using this 65.52 into minus 1 this is minus class into 0.49 then I will be using this but λ i 0 the whole value will become 0 for all this value will become this will become 0, so only just to this to tuple we will consider, so this is what I got w 1 value.

Similarly, I will find w 2 value for w 2 value I will be using these two w 2 x 2, so I will be using these two values and I got this is my w 2 value.
(refer time: 18:48)

Now so once I know the w value, I can find out my b value. So, what is b 1? b 1 is 1- W dot x 1 and b 2 is 1 minus W x 2, so I found this is my b 1 this is my b 2, so both b does one only one b is not it. So, I will take the average of that since both are same values so I got b = 7.93, so I got W value, I got b value, I have all the values I got.
(refer time: 19:10)

Now I can find out my MMH the whole function, what is my function, my function is W x 1 + W x 2 + b = 0, so this is my W x 1 this is my W x 2 + b = 0, so this is my MMH. So, when once my MMH is there nothing is left just if you want to given any test data, I will be able to tell it which class does it belong just I will be using the MMH. So, how we have already seen how to find out the class of any tuple testable does the delta of X is W of X W dot X + b.

So, what is my X value, X value is point this is my x value is not it so what is my W value this is 0.64 9.32 - 0.61 so - 6.64 into 0.5 - 9.32 into 0.5 plus this b value 7.93, so I got this minus 0.05. So, that means it is a negative value, negative value means it falls below the hyper plane, it falls in a negative class. This implies that the test data falls on or below the MMH and the SVM classifies that X belongs to class level minus.

This is how we do, how we have found out once we found out the lagrangian maximum margin hyper plane just maximum margin hyper plane is known then nothing is left, like then we will have to given any testable just W X + b just find out W X + b but sign it gives if it is positive above, if it is negative below.
(refer time: 20:55)

So, now this is till now we have done for binary classification, let us see for multi class classifications. In the discussion of linear SVM we have limited our discussion to binary classification of two classes. The discuss linear SVM can handle any dimension for n greater equals to zero what we have discussed that the technique the matter what we have is discuss for linear SVM that can also handle for n greater equals to 0.

Till now our n = 2 binary classification. Now, we are this will discuss a more generalized linear SVM to classify n dimensional data belonging to two or more classes. It may be two classes or more class. So, there are two different ways how we will see it one is of course that will come different there are two when there is classes there are more than two classes there are two different possibilities.

One is all classes were pairwise linearly separable fear wise linearly all the classes. Suppose there are two class three classes one class I am writing is plus minus and into say these are the three classes. All these classes are linearly pairwise linearly separable, plus and minus a linear pairwise linearly separable, plus and into is pairwise linear separable, minus and into is pairwise linearly separable, plus and into is pairwise linearly separable, all this class are pair wise linearly separable linearly.

What is linearly separable? What is non-linear separable? I have discussed in my first class on SVM. So, there one is all are pairwise linearly separable and another one is there is non linearly separable, there are overlapping. Some I cannot separate it linearly using a hyper plane when I call it a linear event using a hyper plane, I can just separate it into two classes when I cannot separate it this way that is called non-linearly separable.
(refer time: 22:40)

So, there are two different possible days if the classes are pairwise linearly separable then we extend the principle of linear SVM. If the if it is pairwise linearly separable then we can use whatever technique we have learned for linear SVM binary classification same technique we can use here. If it is not linearly separable definitely, we cannot use that that we will see later what we will do in that case.

So, there are two set resists if there are pairwise linearly separable one strategy is called one versus one such as that is OVO strategy another is one versus all statistic is OVA strategy, OVO and OVA.
(refer time: 23:21)

And over strategy we have to find MMH for each pair of classes very easy. So, there are three class plus minus into I will have to find out MMH for each class. So, class minus into I will find a MMH for plus and minus plus and into then minus and into, all the classes I happen all different combination. So, if there are n classes then n C 2 classifier is possible, so total here if there are 3 classes I found how many classifieds three classifiers.

So, if there are n classes here there are 3 classes. So, if the n classes to totally how many classified there will be n C 2 classifiers. For an example in the figure if you see plus minus n into so here there are three classifiers for plus n minus n into plus and into total there are three classifiers. So, let H x y denotes MMH between class level x and y in linear linearization means we have to find out the MMH.

So, there are total three classifiers means we will have to find out total three different MMH. And another example you can think say for four classes if there are four classes so that how many classifier there will be four C 2 classifiers.
(refer time: 24:44)

So, with over strategy we test each of the classifier in turn and obtain del j i that is the sign of the MMH between the jth and the ith class for test data X given any test data suppose this is my test data. What I will do? Some suppose consider this example here example only where there are three classes three classes means I got three C 2 so I got total three classifiers. So, in over strategy what I will do? I will be testing my test data with all the three classifiers.

So, with all the three classifiers and I will find a sign of the when I will find out how this is remember this is how I find out the class this is del of X and I found find a sign it belongs to which class, similarly I will find a del of X for all the classifiers using all the classifiers. So, I will find del of X using all the classifiers this I am instead of writing this del of X I am giving a different thing means which two classes I am comparing del of j and i.

So, based on this I will find as sign which is belongs to which class. So, now if there is a class i for which del of i j for all j gives the same sign then unambiguously, we can say X is in class i. So, what happens suppose this is my test data I checked with this classifier first this classifier suppose I found that with this class where I found with some magnitude, I found that it falls belongs to the minus category minus class.

Then again what I do again I will test this with this classifier, again I will get some values either this or that. There has to fall either this or this but the classification will not be proper because this the item this data actually does not fall in this class. So, classification I will not get a very good magnitude but it will fall in one of this class I will get a sign accordingly. Then again what I will do again I will check with this classifier minus and into here again I got minus class.

I will see that it will fall below this and it below I will see that I got is a sign as minus. So, here this is if this is a class i for which del of j i for all J that J not equals to i, not equals to i means for all plus and into gives the same sign I tried with plus I tried it into I got the same sign when I tried with i n plus I got minus when I tried i then into I got minus I got gives the same sign then unambiguously we can say x is in class I, this is the OVO strategy but it has OVO strategy has some cones.
(refer time: 27:38)

OVO strategy is not useful for data with a very large number of classes. Why large number of classes? As the computational complexity increases exponentially with the number of classes because if the number of classes are more see how many classes how many classifier will be n C 2 classifiers, when n becomes more you can see how many classifiers will be there, more number of classifiers that many number of times we will have to check if for each test data.

As an alternative to OVO strategies is OVA one versus all strategy has been proposed. So, what is OVA strategies? In this approach we choose any class say C 1 and consider that all other tuples classes belong to a single class. What we do says, suppose there are 3 plus minus into also is just one this and this I will consider another class. So, simple binary classification this is one and rest is other is one class.

Now again I will say is minus and plus and into is another driven class, now again also into and plus and minus is another one class, got it. So, how many classifiers I needed? So, if there are 3 classes I needed 3 classifiers, if there are 4 classes I will be needing 4 classifiers instead of n C 2 in the OVO strategy.
(refer time: 28:55)

So, this is therefore transformed into a binary classification problem using the linear as we have discussable and we can find a hyper plane. Let the hyper plane between C i and the remaining classes be MMH i. The process is repeated for each C i as I told for each plus minus into and getting the different MMH, in other words over strategies we get total k classifiers if the k

classes will get k classifiers unlike the other one where we get k classes means k C 2 classifiers which is a huge number.
(refer time: 29:33)

So, the unseen data X is then tested with each classifier. So, this unseen data so we have total one is minus and a plus and into another is plus minus and into another is into minus n plus so here is one classifier here is one classifier and raise one classifier. So, we will test it for on using all these three classifiers, the unseen attacks is then tested which each classifier so, obtained. Now this del of j X be the test result with from for each value for each classifier I will be getting this del of X is not it.

For each classifier I will be getting a del of X which has the now among this which has the maximum magnitude for this I will be getting a del of X with a particular magnitude, for this I will be getting a del of X with a magnitude, for this I will be getting a del of X with a particular magnitude the one which will give me the maximum magnitude that will be the class for the test tuple.

The X i will be the test result for MMH i which has the magnitude maximum magnitude of the test values. That is del of j is the maximum of this for all i, thus X is classified into class C j whichever gives us the maximum value magnitude for this del of j that is the class of the test number. So, note that the linear SVM that is used to classify multi-class data fails if all the classes are not linearly separable.

This we can do either OVO strategy or OVA strategy, both we can do if the datas are linearly separable, mostly uses OVA strategies OVA not OVO but then we can do only the datas are linearly separable. If one class is linearly separable to remaining other class and a test data belongs to that particulars then only it classes accurately. If we have many classes out of them one class is linearly separable to remaining other classes.

And now if we are given a test that as belongs to this class which is linear separable then we can accurately classify but there is some another class say X which is not linearly separable to other classes and my test data belongs to that group then I will not be able to classify it accurately, then my this strategy will not work properly it will give a poor result.
(refer time: 31:57)

See you can see it here, see the figure if it is possible to have some tuples which cannot be classified any of the linear SVMs like these things it cannot be classified to any of the classes because they are not linearly separable. There are some tuples which cannot be classified unambiguously by neither of the hyper planes, unambiguously we cannot do. When we are trying with different classifiers sometimes it is getting this sometimes.

It is giving this anonymously we cannot find out this class this tuple belongs to this class only. When it will happen? When there are some tuples which cannot be linearly separable all this tuple may be due to noise errors or data and are not linearly separable, sometimes this table may be due to noise or some errors and sometimes because the data is are not linearly separable. If the datas are not linearly separable then we will have to think of some other way. So, we will be discussing next in our next lecture.
(refer time: 32:52)

So, in this lecture we derived the equation of optimal hyper plane using the lagrangian multiplier, we have seen that so, we also learned to classify given a test data we will element method through an example is not it. Then also we have seen how to extended these two class linear SVM to multi class classification problem we have seen that and we have seen that if their datas are not linearly separable then linear SVM will not work for multi-class.

Then in that case what we will do for multi glass or two class whatever it is if the data’s are not linearly separable it will not give us good result. So, in that place we will have to learn we will have to use non-linear SVM, in our next lecture we will learn this non-linear SVM.
(refer time: 33:33)

So, these are the references and thank you guys.