Deep Learning
Prof. Prabir Kumar Biswas
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture - 11
Support Vector Machine - I

Hello, welcome to the NPTEL online certification course on Deep Learning.

(Refer Slide Time: 00:33)

CONCEPTS COVERED

Concepts Covered:

❑ Linear Discriminator

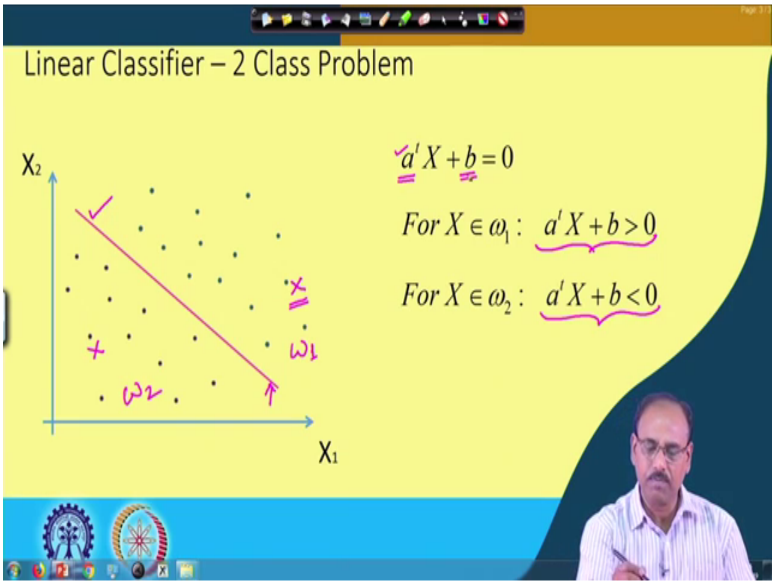
❑ Perceptron Algorithm

❑ Support Vector Machine (SVM)

In our previous lecture, we had talked about linear discriminator and we had also talked about the perceptron algorithm in which our linear using which a linear discriminator can be designed. Today we are going to discuss about the Support Vector Machine and if you remember that the classes that we have considered in the previous day are actually 2 classes. So, we wanted to have a linear discriminator which discriminates the vectors belonging to two different classes. And, when you talk about support vector machine we will continue with 2 classes initially, but later on we will move to multi class classification problems.

So, before we go for support vector machine I will just quickly recapitulate what we have done in the previous class in the linear discriminator.

(Refer Slide Time: 01:32)



Linear Classifier – 2 Class Problem

$a^t X + b = 0$

$For\ X \in \omega_1 :\ a^t X + b > 0$

$For\ X \in \omega_2 :\ a^t X + b < 0$

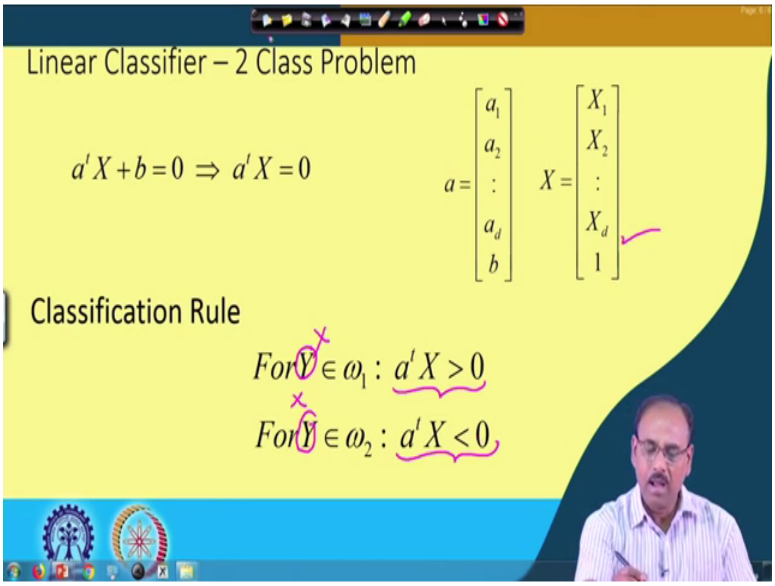
So, we what we had done in the previous class is we have taken 2 sets of feature vectors so, this is of set of each a vector which belong to class omega 1 and we have taken another class another set of feature vectors from class omega 2. And, then what we tried to see is we assumed that these 2 classes these 2 states of feature vectors are linearly separable and assuming linear separability, we have tried to find out a linear boundary or a hyperplane which separates these 2 classes of feature vectors. So, find that an equation of such a linear boundary will be given by this a transpose X plus b equal to 0, where you find that this vector a is a vector which is orthogonal or normal to the separating plane.

So, the vector a will be like this so, this will be the direction of vector a and as we have assumed the existence of a separating boundary which is linear; we know from our school level mathematics that such a linear boundary divides the feature space into two half spaces. One of the half space is positive half space, the other half space is negative half space. So, for any feature vector X which belongs to positive half space I must have a transpose a X plus b which is greater than 0 and for every feature vector lying on this planes.

So, if I take a feature vector lying on this plane for this feature vector a transpose X plus b will be equal to 0. So, as we have taken a number of feature vectors from the 2 classes omega 1 and omega 2, if I take any vector X from the class omega 1. So, this is the set of vectors belonging to class omega 1 and this surface this being the linear boundary having equation a transpose X plus b equal to 0 for every X belonging to class omega 1 which are my training vectors this condition must be satisfied that a transpose X plus b have to be greater than 0.

In the same manner, if I take a feature vector X from class omega 2 where this feature vector X falls on the negative side of the linear boundary this condition that a transpose X plus b less than 0 must be satisfied. So, for this surface for this linear boundary which satisfies this equation a transpose b equal to 0 as we have said that the vector a is orthogonal to the surface. So, if I modify vector a; that means, the orientation of this linear boundary will change and the value of b which is nothing, but a bias it decides the position of this separating plane in the feature space. So, a gives you the orientation and b gives you the position.

(Refer Slide Time: 05:02)



Linear Classifier – 2 Class Problem

$$a^t X + b = 0 \Rightarrow a^t X = 0 \qquad a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_d \\ b \end{bmatrix} \qquad X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_d \\ 1 \end{bmatrix}$$

Classification Rule

$$\text{For } \overset{x}{Y} \in \omega_1 : \underbrace{a^t X > 0}$$

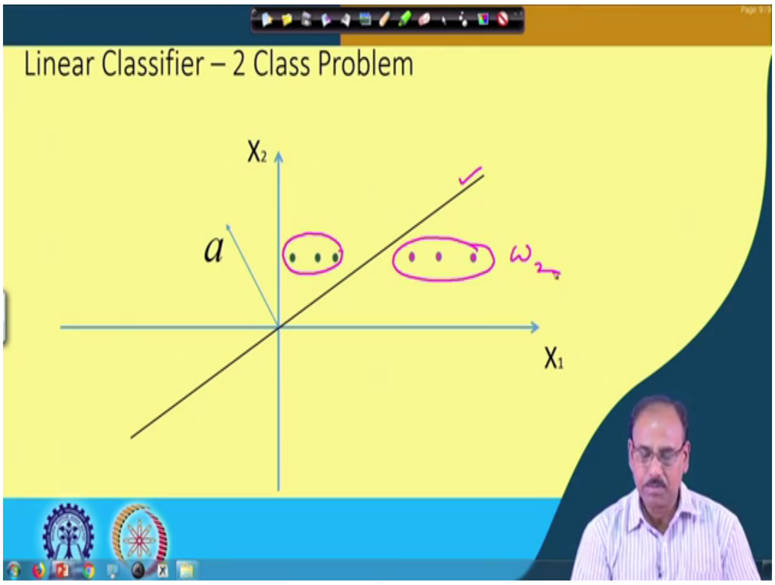
$$\text{For } \overset{x}{Y} \in \omega_2 : \underbrace{a^t X < 0}$$

Now, given this I can also represent this equation a transpose b a transpose X plus b equal to 0 in an unified form a transpose X equal to 0. And, for writing this equation a transpose X plus b equal to 0 in the form a transpose X equal to 0 I have to do certain modifications. So, what are those modifications? I have to modify a like this, that now this modified a contains all the previous components a 1 to a d of my initial feature vector the initial solution vector a. And, this bias term b is also now included in the same vector a in this modified vector a. And, in order to do this the feature vector X has to be modified as all the components of X that is X 1 2 X d are remaining as it is.

Now, what I have to do is I have to append an additional component to X which becomes 1. So, with this modification your a transpose X plus b equal to 0 now gets modified to a transpose X equal to 0. So, that my bias term b is included in the solution vector a. So, you find that the implication of this equation is now this separating plane always passes through the origin in my d plus 1 dimensional space. In the earlier case depending upon the value of b the separating plane might have been anywhere within my feature space. But, in this modified form as I am increasing the dimension of the feature vector by 1 in this modified feature space the separating plane always passes to the origin.

So, given this my situation will now be of this form, the classification will rule now will remain that for every X sorry this should be X, for every X belonging to class omega 1 I must have a transpose X greater than 0 so, this Y have to be X right. For every feature vector X in this modified form taken from plus omega 1, I must have a transpose X greater than 0 and if it is taken from class omega 2 I must have a transpose X less than 0. So, for correct classification the classification rule remains the same and I can do another modification that is.

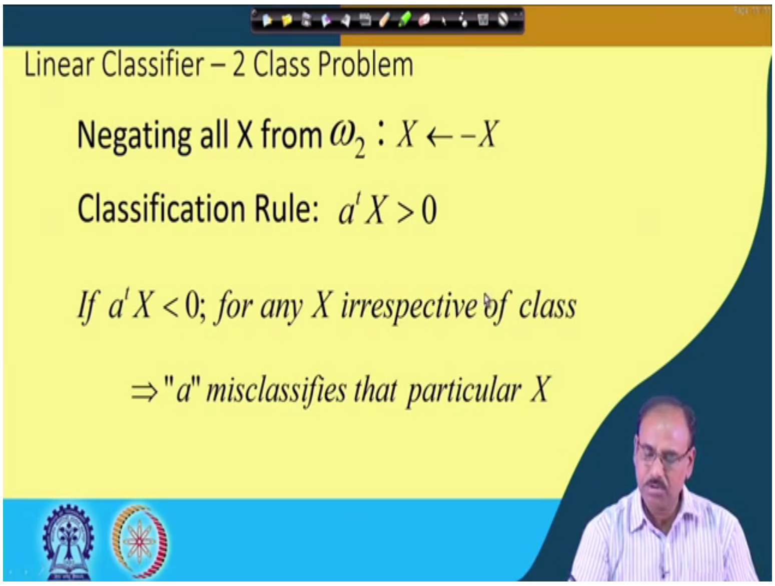
(Refer Slide Time: 07:46)



Linear Classifier – 2 Class Problem

So, what does it mean is this that coming over here you find that as one of the components of the feature vector X I have made equal to 1. So, in the modified form the feature vectors will appear like this. So, if this is the X 1 component and this is the X 2 component, X 2 component has been made equal to 1. So, this will be the arrangement of which are vectors in this case and a being the orthogonal to my separating plane, these vectors which belong to class omega 1 appear in the positive half space of the of the separating plane. And, these vectors which belong to class omega 2 appear in the negative half space in my feature space.

(Refer Slide Time: 08:40)



Linear Classifier – 2 Class Problem

Negating all X from ω_2 : $X \leftarrow -X$

Classification Rule: $a^t X > 0$

If $a^t X < 0$; *for any X irrespective of class*

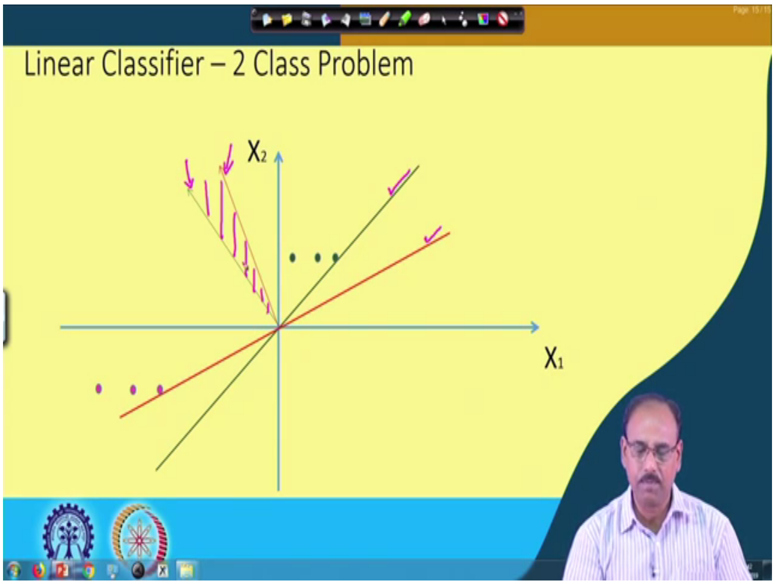
$\Rightarrow$ "*a*" *misclassifies that particular X*

I grew another modification that is all the feature vectors which come from class omega 2 I negate them ; that means, a feature vector X if it is taken from class omega 2 I make it minus X. What is the advantage? By negating all the feature vectors coming from omega 2 negative and negating them my classification rule becomes same irrespective of whether X is taken from class omega 1 or X is taken from class omega 2. Because, in every case my classification rule becomes a suppose X greater than 0.

And, if I find that given a solution vector a for any X irrespective of whether this X belongs to class omega 1 or X belongs to class omega 2, if I find that for any such a X a transpose X is less than 0 immediately I can say that this solution vector a misclassifies the corresponding X. And, whenever there is a misclassification I must try to update a such that the modified a or update a will correctly classify X; let us see how we can do it. So, my situation is something like this now, that in the previous case you find that all these feature vectors belonging to omega 2; they were on the negative side of the separating plane.

Now, after negation that negated feature vectors belonging to class omega 2 now comes over here. And, you find that after negation all the feature vectors, the negated feature vectors belonging to class omega 2 and also the original vector feature vectors belonging to class omega 1, all of them fall on the positive side of the separating plane. Now, you find that I can observe one more thing that is what is the limit of this separating plane or what is the limit of the solution vector a. So, you find that if I rotate the separating plane in the anti clockwise direction then the limit to which I can rotate this is given by this.

(Refer Slide Time: 10:59)

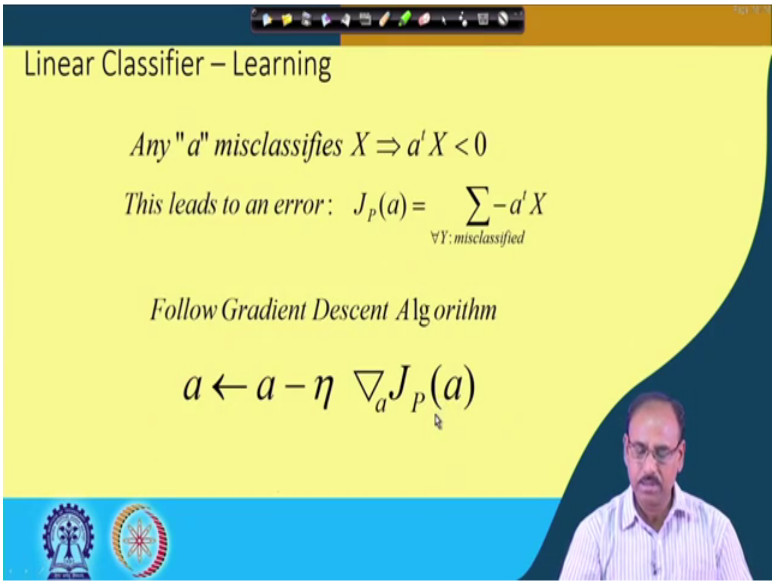


Linear Classifier – 2 Class Problem

Because, if I rotate it further in the anti clockwise direction then this feature vector belonging to class omega 1 is going to be misclassified. In the same manner if I rotate it in the clockwise direction then this is the limit that I can have because, if I rotate it further in the clockwise direction then this feature vector belong to class omega 2 that is going to be misclassified. So, these two this one this position and this position gives me a limit of the separating plane.

And, as I have limit on the separating plane, in the same manner I can have limits on the corresponding solution vectors. So, you find that as this position is the limit of the separating plane, the corresponding limit on the solution vector is this which is orthogonal to the separating plane. Similarly, here the corresponding limit on the orthogonal or the solution vector is this. So, it clearly says that I must have an aid which correctly classifies all the training vectors must be within this region which is my solution region. So, the approach for designing a linear classifier should be such that I must get a solution vector lying within this solution region.

(Refer Slide Time: 12:31)



Linear Classifier – Learning

$Any\ "a"\ misclassifies\ X \Rightarrow a^t X < 0$

$This\ leads\ to\ an\ error:\ J_p(a) = \sum_{\forall Y: misclassified} -a^t X$

$Follow\ Gradient\ Descent\ Algorithm$

$$a \leftarrow a - \eta\ \nabla_a J_P(a)$$

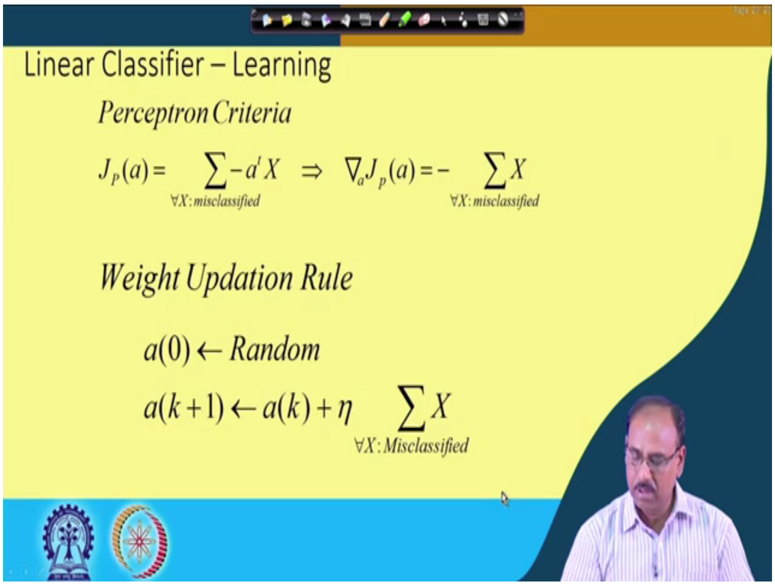
So, in order to do this what we can do is for every X which is misclassified as I know that for every X which is misclassified by a, I should have this condition that a transpose X less than 0. You remember that all this X is that we are talking about, all these feature vectors X that we are talking about these are all training vectors; that means, for all the vectors I know to which class they belong. So, as we have said so, far that given any a if I find that a transpose X becomes less than 0; that means, that a misclassifies that X and this misclassification leads to an error.

So, I can have an error measured for this misclassified sample which I will put as minus a transpose X. So, as a transpose X is less than 0 so, minus a transpose X is positive; that means, if I have a misclassified sample that leads to a positive error. So, whenever a correctly classifies all the samples then this error will be equal to 0. So, what I do is for given a you identify all the feature vectors which are misclassified; that means, all the feature vectors for which a transpose X becomes negative. And, then you define an error function which is called the perceptron criteria function.

So, I write this as J P a which is a function of now the solution vector a which is as sum of minus a transpose X and this summation has to be taken over all X which are misclassified. And, once I have this error measured then I can modify a following gradient descent algorithm, we will discuss more about gradient descent algorithm later. So, this gradient descent algorithm says that I have to shift a in that direction of the negative gradient.

So, for updation of a; I have this updation rule that a gets a minus gradient of J P a and this gradient is scaled by a scale factor eta which is known as rate of convergence. So, this will be my weight updation rule using gradient descent procedure. So, in this particular case where, I have this perceptron criteria function J P a given by minus a transpose X for some of that for all X which are misclassified. So, using this error function the gradient descent procedure now becomes.

(Refer Slide Time: 15:26)



Linear Classifier – Learning

Perceptron Criteria

$$J_p(a) = \sum_{\forall X: misclassified} -a^t X \Rightarrow \nabla_a J_p(a) = -\sum_{\forall X: misclassified} X$$

Weight Updation Rule

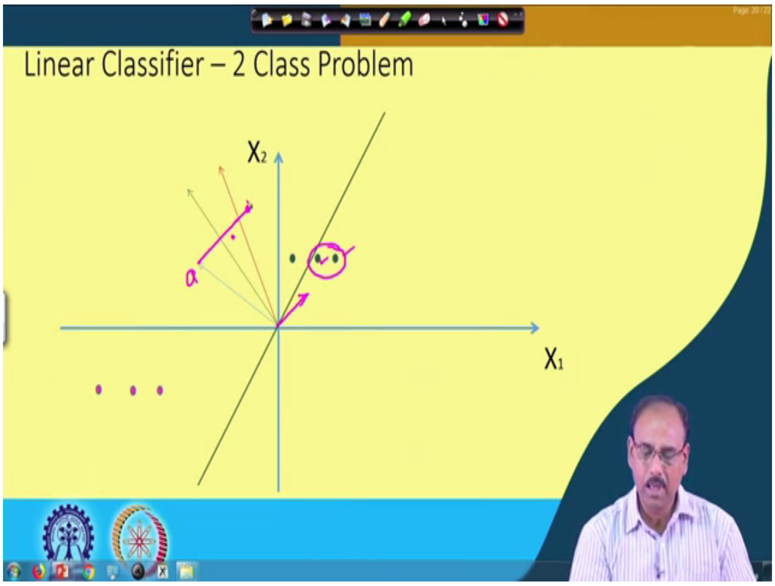
$$a(0) \leftarrow Random$$

$$a(k+1) \leftarrow a(k) + \eta \sum_{\forall X: Misclassified} X$$

If you take the gradient of J P a, the gradient of J P a becomes minus sum of X for all X which are misclassified. And, accordingly my training or learning algorithm will be like this that initially you choose weight vector a 0 at random and then I will go on updating this weight vector a iteratively. So, in any case iteration if a k is the weight vector using this a k you try to identify all the feature vector X which are misclassified.

And, once you identify all such which are vectors which are misclassified then for the next iteration or the next updated weight vector a k plus 1 I can get it from a k by modifying a k as a k plus 1 gets a k plus eta times sum of X for all X which are misclassified. So, this is my weight updation rule, let us say with diagram what does it mean.

(Refer Slide Time: 16:38)

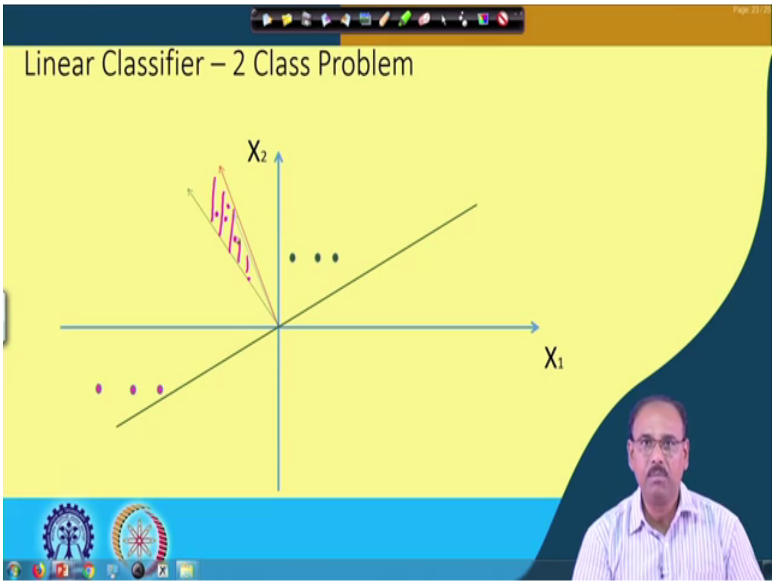


Linear Classifier – 2 Class Problem

So, as we have seen before that given these two sets of vectors belonging to class omega 1 and omega 2, I have the region solution region which is this so; that means, any solution vector must lie within this region. So, what I do is initially let us assume that we have a separating plane which is given as this. And, here you find that this separating plane misclassifies these two samples which belong to class omega 1 and the weight vector a is this which is; obviously, outside the solution region. So, I have to update this vector a by adding to it, the sum of these two misclassified vectors and when you add the sum of these two misclassified vectors is obviously, in this direction.

So, a has to be moved in this direction and if I have sufficient eta, eta is proper then possibly we will stop within the solution region and I get the solution. But, if eta is large then we will cross the solution region and will move somewhere over here. So, here in this case after adding modifying this a using some of these two misclassified samples suppose the next separating plane comes out to be this.

(Refer Slide Time: 18:09)

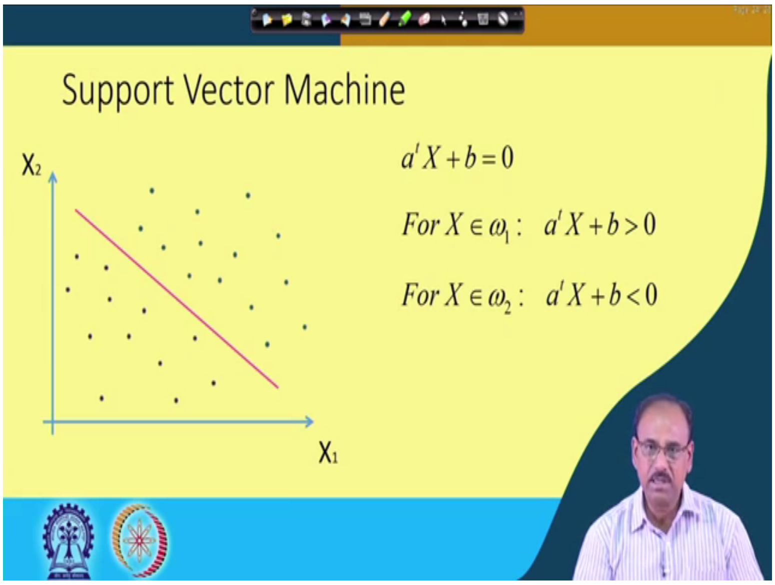


Linear Classifier – 2 Class Problem

And, again this separating plane as you see misclassifies these two samples and the corresponding solution vector a 1 is over here. So, again to a 1 you add some add some of these two vectors and some of these two vectors is in this direction. So, a 1 has to be moved in this direction. So, at the next level my separating plane will be somewhere like this by updating a 1. And, now you find that I have a vector which falls within this solution region and this is my a 2.

And, as it falls within the solution region at and it correctly classifies all the samples whether it belongs to class omega 1 or belongs to class omega 2; so, I am satisfied with this solution vector. So, this is the approach for designing a linear classifier, now as we have seen that as the solution region is this. So, any vector within this region should satisfy my purpose, but if the solution vector comes very close to this solution boundary possibly that is not a good solution because, it is prone to error. So, I would like to have a vector which is well within this solution region so, that the classification my classifier becomes very very robust. So, let us see how we can do it.

(Refer Slide Time: 19:48)



Support Vector Machine

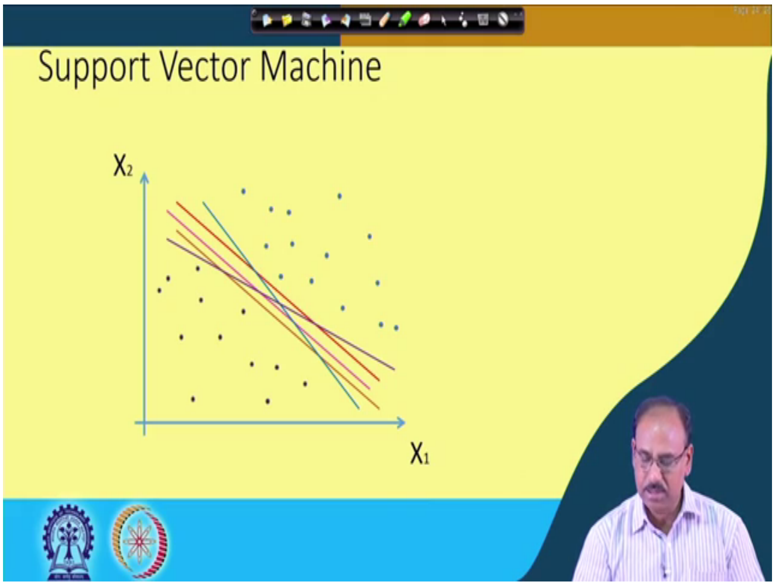
$a^t X + b = 0$

$For\ X \in \omega_1 :\ a^t X + b > 0$

$For\ X \in \omega_2 :\ a^t X + b < 0$

So, what I mean by this is say given all these different feature vectors coming from two different classes and earlier as we said earlier that my solution classification rule is a transpose a X plus b be greater than 0, if X belongs to class omega 1. And, a transpose X plus b is less than 0 if X belongs to class omega 2 and this is one such separating plane which satisfies this criteria, but is it unique obviously, not let us see.

(Refer Slide Time: 20:23)

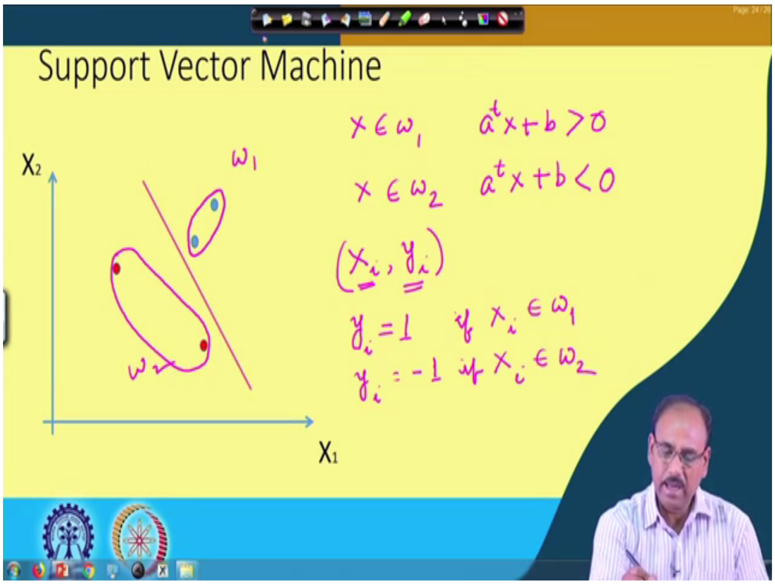


Support Vector Machine

So, this is one such separating plane that we have seen which satisfies this region, I can have another separating plane which is given by this blue line, that also satisfies this criteria. This is the separating plane which also satisfies this criteria; this is the separating plane which also satisfies this criteria. So, I have infinite number of such possibilities.

So, I have to identify that out of all these different possibilities which one is should be the preferred solution and that is where the support vector machine comes into picture.

(Refer Slide Time: 21:05)



Support Vector Machine

$x \in \omega_1 \quad a^t x + b > 0$

$x \in \omega_2 \quad a^t x + b < 0$

$(\underline{x_i}, \underline{y_i})$

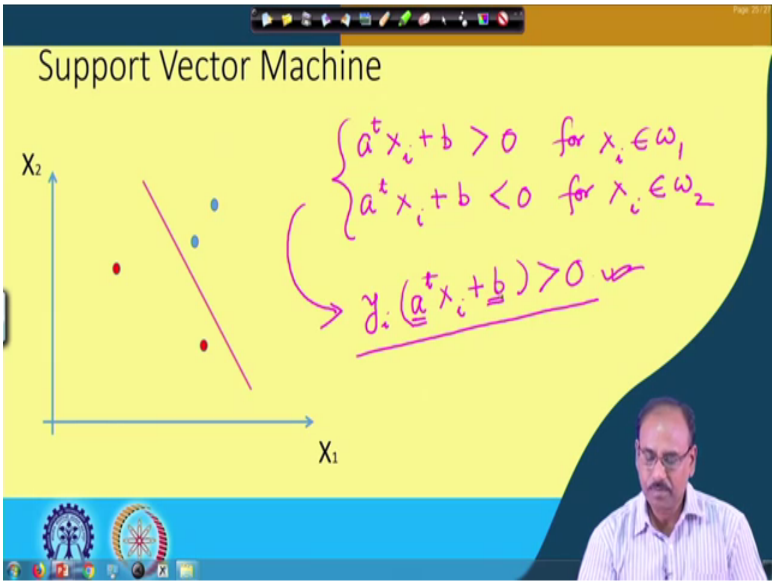
$y_i = 1$ if $x_i \in \omega_1$

$y_i = -1$ if $x_i \in \omega_2$

So, to illustrate this let us take a very simple case. So, the case is like this I assume that these are the vectors which belong to class omega 1 and these are 2 vectors which belong to class omega 2. And, I have a separating plane which separates in these 2 classes and now as I have shown previously that for any X belonging to class omega 1, my correct classification criteria is a transpose X plus b greater than 0. And, for X belonging to class omega 2 I had a transpose X plus b less than 0.

Now, I can take another strategy; let us assume that every training vector is given as a pair ; that means, along with the training vector we also have its class level right. So, I can put it this way that training vector X i is given as a pair X i Y i where, this y i indicates that what is the class or to which class X i belongs. So, I will put y i as plus 1 if X i belongs to omega 1 and y i will be is equal to minus 1 if X i belongs to omega 2.

(Refer Slide Time: 22:54)



Support Vector Machine

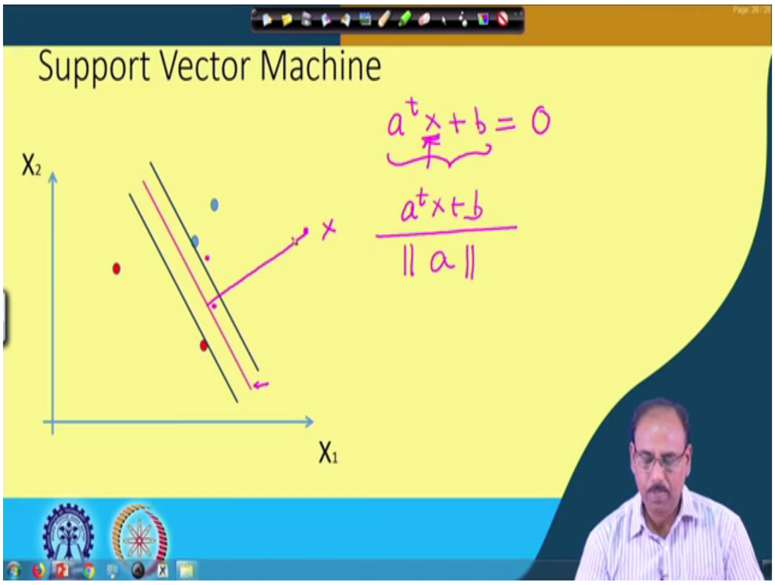
$$\begin{cases} a^t x_i + b > 0 & \text{for } x_i \in \omega_1 \\ a^t x_i + b < 0 & \text{for } x_i \in \omega_2 \end{cases}$$

$$\underline{y_i(\underline{a}^t x_i + \underline{b}) > 0}$$

So, if I assume this then you find that both of these conditions that a transpose X i plus b greater than 0 for X i taken from class omega 1 and a transpose X i plus b less than 0 for X i taken from class omega 2 both of them can be written in a single form that is y i times a transpose X i plus b, it has to be greater than 0. So, this becomes my unified representation and if you compare this with what we discussed previously, that we negated all X and had an unified representation all X taken from class omega 2. And, had an unified representation or unified classification rule of a transpose X greater than 0 for correct classification which is exactly same as this.

It is just another way of representation that you multiply by y i a transpose X i plus b where, y i is plus 1 if X is taken from class omega 1 and y i is minus 1 if X i is taken from class omega 2. And, by doing that I get an uniform classification role that y transpose y i into a transpose X plus b will be greater than 0; whenever X i is correctly classified by this vector a and the offset of the bias term b ok. So, now as we have seen previously that I can have different options of the separating plane and what I have to do is I have to choose out of all those options which is the correct option.

(Refer Slide Time: 25:10)



Support Vector Machine

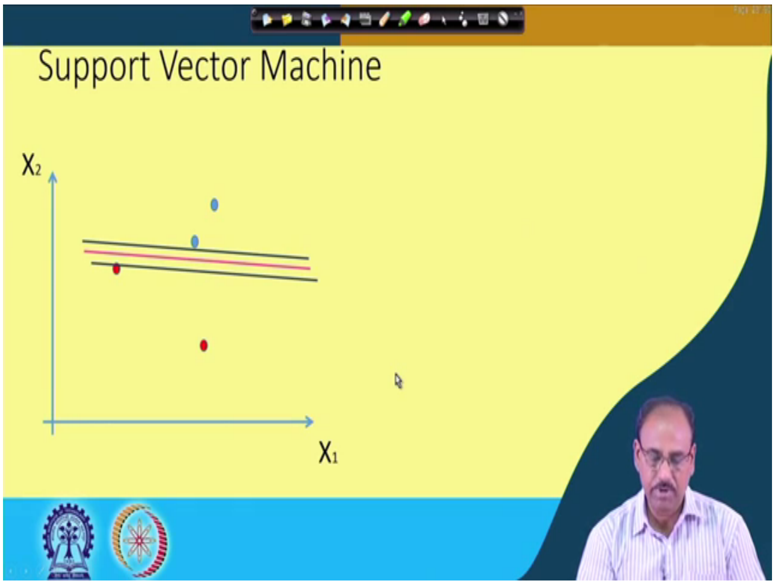
$$\underbrace{a^t x} + b = 0$$

$$\frac{a^t x + b}{\| a \|}$$

So, if I take this particular separating plane then you find that my margin is given by this you. Remember one more thing that when I take a transpose X plus b for any feature vector X, this is an indication of what is the distance of the feature vector X from the separating plane a transpose X plus b equal to 0. So, given over given in this equation, if this is my separating plane my confidence of correctly classifying a feature vector X which is lying over here is more than my confidence of correctly classifying a feature vector over here or directly classifying a feature vector over here.

And, what is this a transpose X plus b? a transpose X plus b as I said it is the distance of vector X from the separating plane a transpose X plus b equal to 0. Or, in other words a transpose X plus b upon mod of a this is the distance of X from the separating plane. So, more the distance more is my confidence that I have correctly classified this sample X. Now, going like this if I take in the feature the separating plane to be this I find that my margin is classification is given by this much; that means, this feature vector the confidence level of this feature vector being classified correctly is this. And, the confidence level of this feature vector pink correctly classified as given by this.

(Refer Slide Time: 27:03)



Support Vector Machine

On the other hand if I take some other orientation of the separating plane said this; now the margin or my confidence level in classification is this much. Let us take another say this one the margin is this. So, you find that for every orientation of the separating plane I have different margins. So, if my classifier is proper or the separating plane is proper or it is robust then I must take that particular separating plane which tries to maximize the margin.

Or, in other words the separating plane which I use for classification of our classification of the feature vector belonging to two different classes, this separating plane must be at a maximum distance from all the feature vectors belonging to belonging to two different classes. That is this separating plane the distance of this separating plane from the feature vectors belonging to class omega 1 and the distance of the separating plane from the feature vectors belonging to class omega 2 from both sides this should be maximized.

So, this is just the introduction of support vector machine and the machine which gives such a separating plane is nothing, but a support vector machine. So, we stop the today's lecture over here and in the next lecture, we will come across that what should be my strategy for designing such a support vector machine.

Thank you.