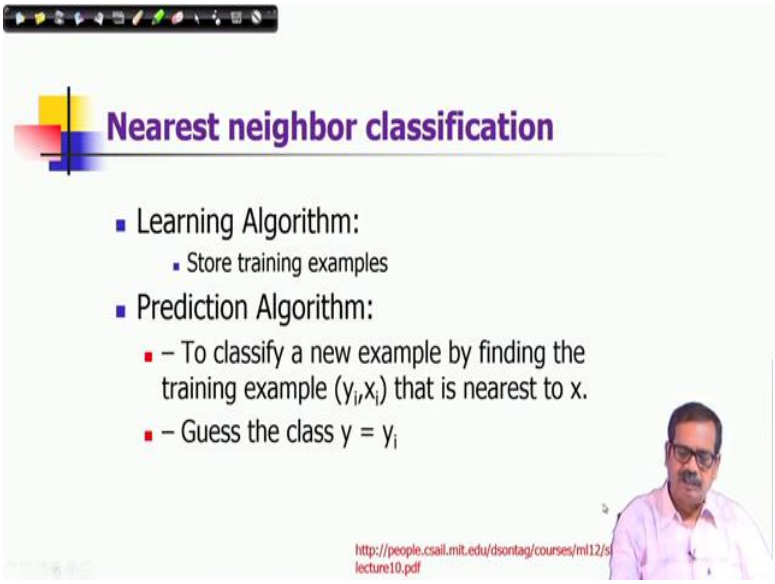
Computer Vision
Prof. Jayanta Mukhopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 48
Clustering and Classification Part - III

We continue our discussion on Classification of data points. And in the last lecture we discussed a classifier called naive bases Bayesian classifier.

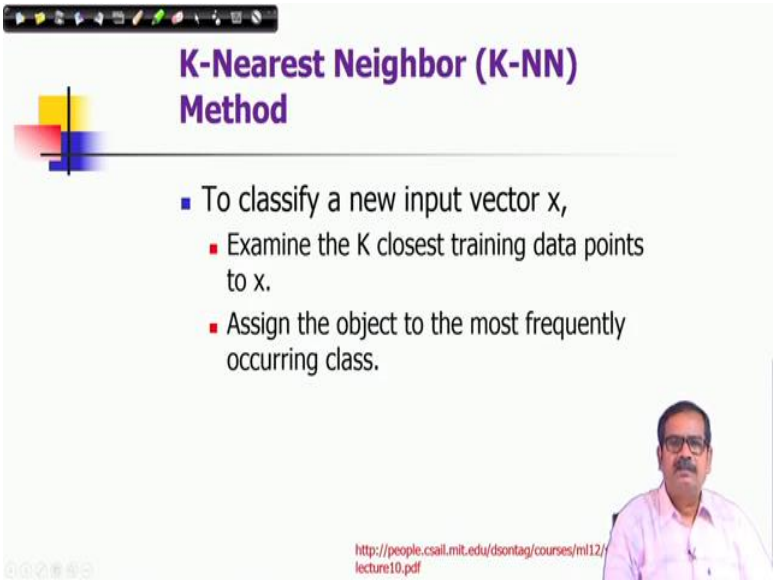
(Refer Slide Time: 00:25)



In this lecture will be discussing about another approach of nearest neighbor classification scheme. And, here the classification approaches quite simple as we can see here. In particular learning algorithm is very simple what you need to do here. You need to simplest towards the training data. In compare to the Bayesian classification you have seen that you need to process the data and perform parametric modeling to get the probability distributions.

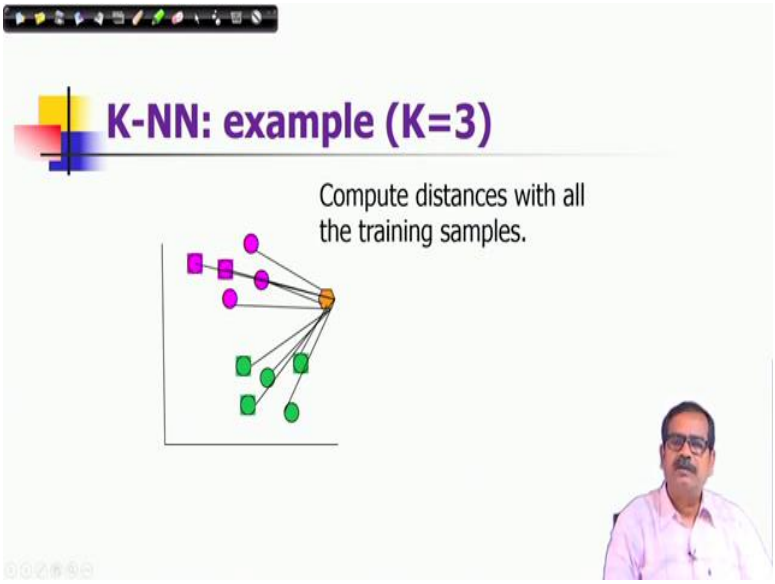
So, but in nearest neighbor classification simply, you store training examples. And then prediction algorithm with goes like this, if you want to classify example by finding the training example that is nearest to x. So, which is one is the nearest, you assign the class to that sample.

(Refer Slide Time: 01:20)



So, one particular method, in nearest neighbor classification is known as K nearest neighbor method. So, instead of observing a single nearest neighbor we observe a set of close neighbors. And, the K most nearest neighbors are called K nearest neighbors. So, in this case to classify a new input vector x; we examine the K closest training data points to x. And then assign the object to the most frequently occurring to us.

(Refer Slide Time: 01:54)

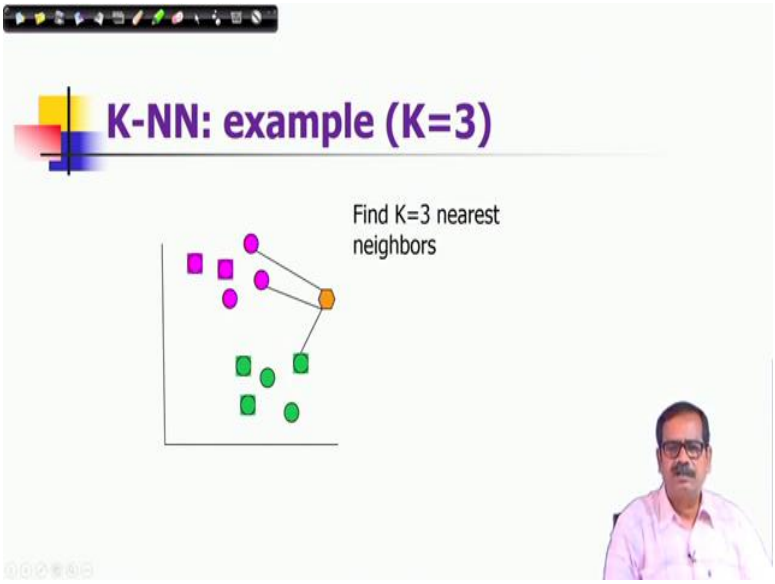


So, here the approach is very simple. Let me illustrate with respect to using this diagram. You can see there are two classes and there are class levels shown by these two colors pink

and green. And suppose you have a query data point it is a 2 dimensional no feature space enquired quarried data point has be shown here.

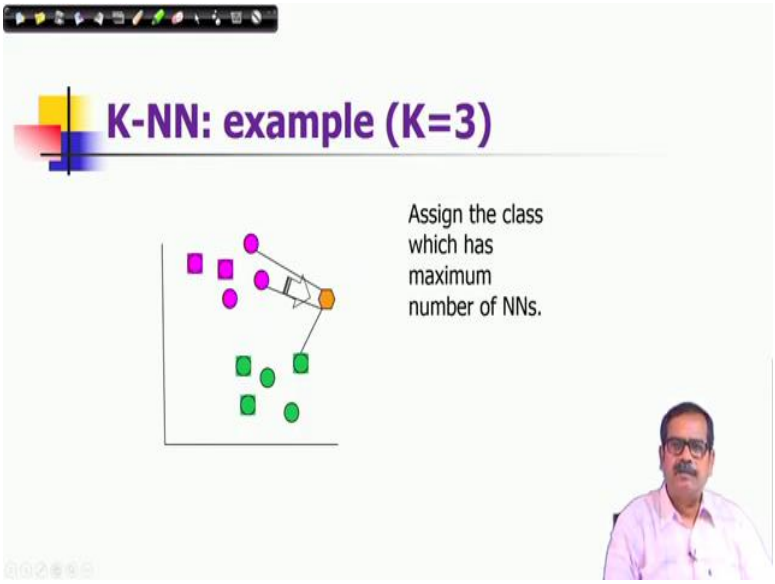
And, then used for computing the nearest neighbors you have to compute distances with all the training samples nearest K close neighbor K most close neighbor. So, you compute distances with all the training sample. And then you can you should solve them and find out three most closest nearest neighbor.

(Refer Slide Time: 02:45)



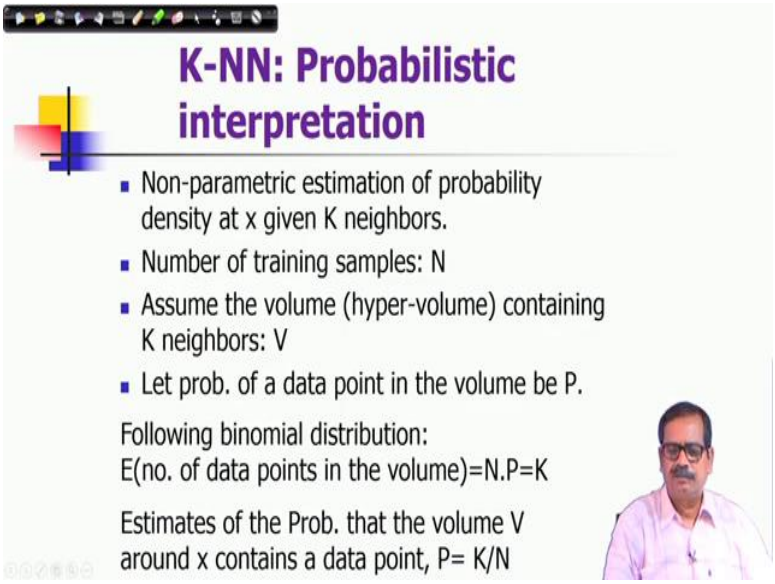
So, you will find that these are the 3 nearest neighbor. So, that is the technical term that is used there.

(Refer Slide Time: 02:56)



And so, and according to the rule what I mention. Assign the class which has maximum number of nearest neighbors of NNs; and we can see that know there are two members from the pink class and one member from the green class. So, we should assign this pink class belong denoting the pink color to that particular query point. So, this is the approach and which is very simple to implement and to understand.

(Refer Slide Time: 03:19)

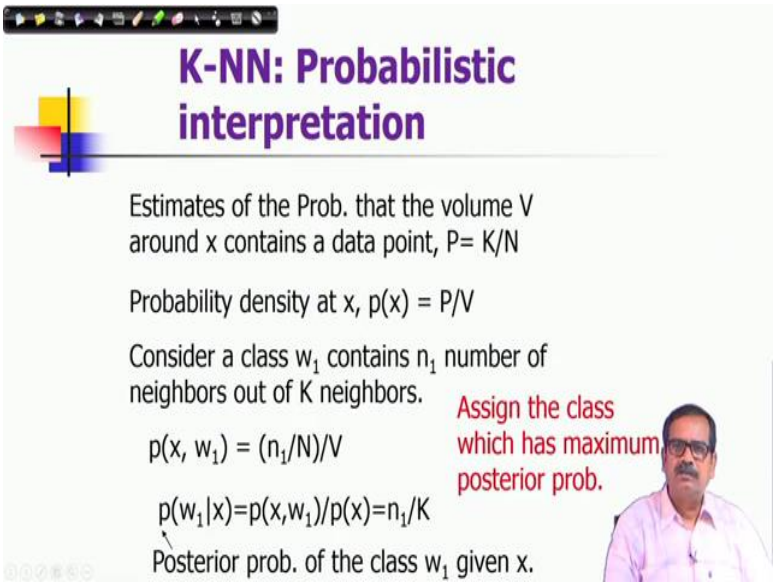


But there are interesting know observation. So, it has a sound theoretical framework though approaches very simple. We can show that actually it is a good relation with the

Bayesian classification rules. So, it is a non-parametric estimation of probability density at x given K neighbors, if I do if I perform these estimations will be doing that. So, suppose there are a number of training samples N. And we assume the volume and in a N dimensional space, we call it hyper value containing K neighbors is V. And let us considered the probability of data point in then in the volume be P. So, following binomial distribution; that means, know how many data could occur.

If it is different kind of its could be modeled, in this fashion that we are sampling the point and with probability P it can occur in the volume V and, if you doing N times this sampling it follows binomial distribution binomial distribution. And, the number of data points in the volume expected number could be N into P and that should be equal to K. So, the estimation of P, is very simple you just get the ratio of K by N or it is the fraction of times it has occurred out of N trails.

(Refer Slide Time: 04:51)



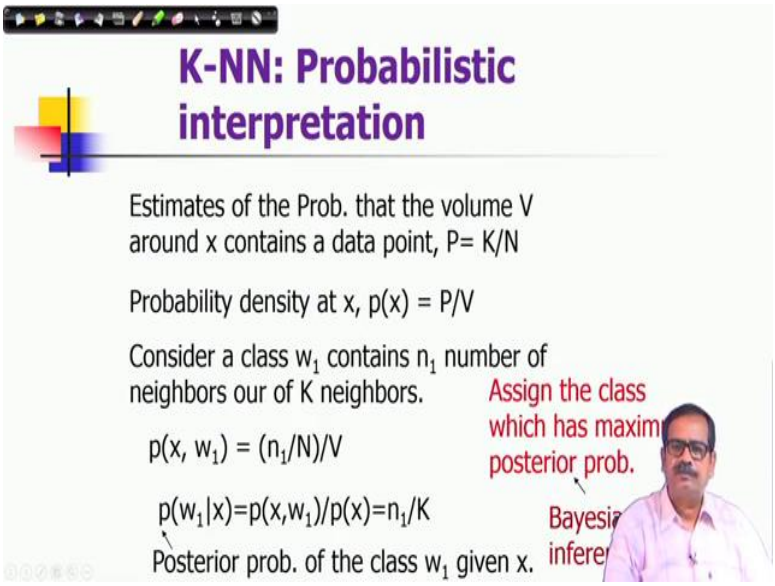
So, this is the is probability that you can estimate here. And then the probability density at x; if I considerates continuous know is space. So, this probability is divided by the volume P that is the know number of data points possible data points here. So, consider a class w 1, which contains n 1 number of neighbors out of K neighbors.

And then the joint probability of x and w 1 will be given by that; that means, first joint probability of interest n 1 has n 1 times that you know it has occurred within that volume. So, it is n 1 by N. And then divided by; that is a density function. So, probability of x given

w 1, which is the posterior probability and which is defined from the Bayesian rule that joint probability divided by probability of x it is n 1 by K.

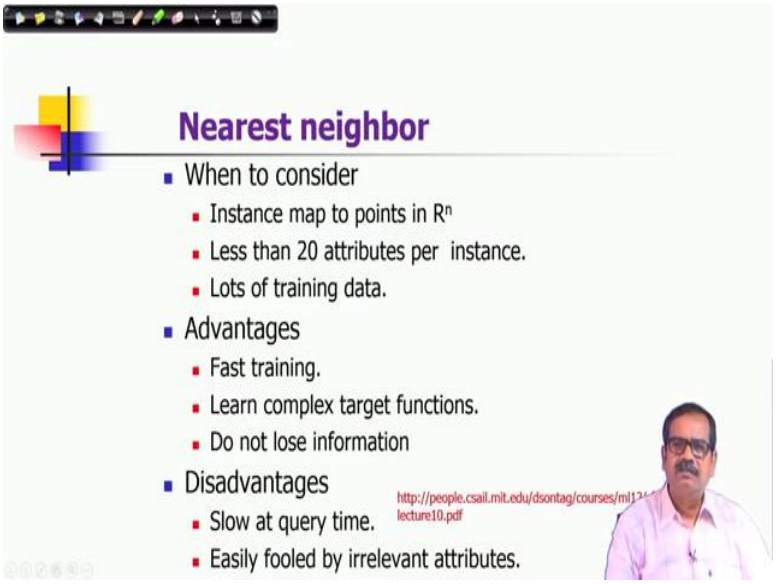
Now, this is the interpretation if you have n nearest neighbors n w 1 out of K neighbors. Then the n 1 by K give the posterior probability of that class given the data point. So, if you assign the class which your for which maximum number of data points of occurred, it is actually in the same as maximizing this posterior probability. So, it is following the Bayesian classification rule as you can see.

(Refer Slide Time: 06:22)



So, this is what and it follows the of Bayesian inferencing rule.

(Refer Slide Time: 06:27)



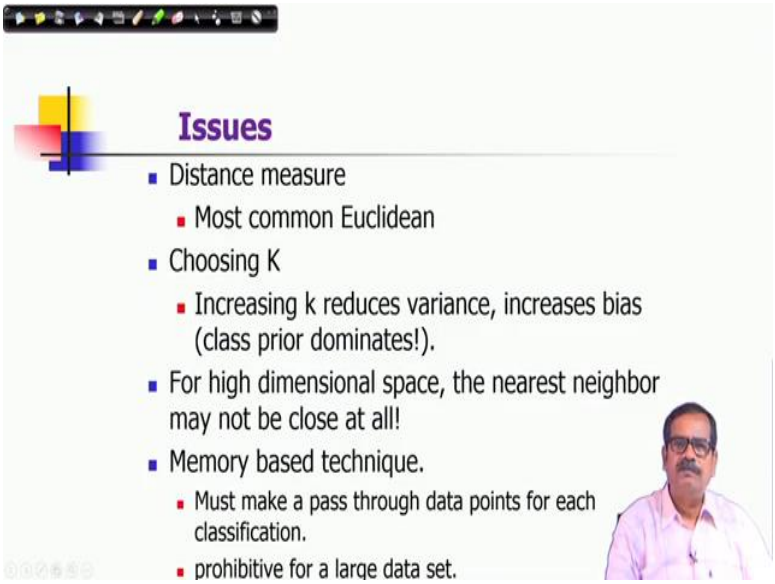
So, when should you consider for nearest neighbor classification? It when the instance maps to points in a in the dimensional space. When you have less than 20 attributes for instance and there are lots of training data. Then those are situations when nearest neighbor classification should be more appropriate. There are advantages as you see that was training what mean it is, it do it means that you have to store the data of course, we will call not storage for that.

So, learn complex target functions the method is very simple, but finally, the function simplicity it is implemented though this method that is not simple. So, if you if you do more detailed analysis, you will be able to understand what are the decision boundaries K nearest neighbors. And those boundaries a quiet complex they are shapes quite complex there not simple.

And it you know it do not lose any information. Disadvantages it is slow at query time it is it is very much competition intensive. You have to compute know all the distances with all the data points. So, which means your query would be very slow. It is linearly proportional with the number of data points.

And if your data point is very large then it is prohibitively very much difficult and easily fooled by irrelevant attributes. So, if there are outliers or rather irrelevant which is not related, but still you are establishing the relationships by nearing relationships.

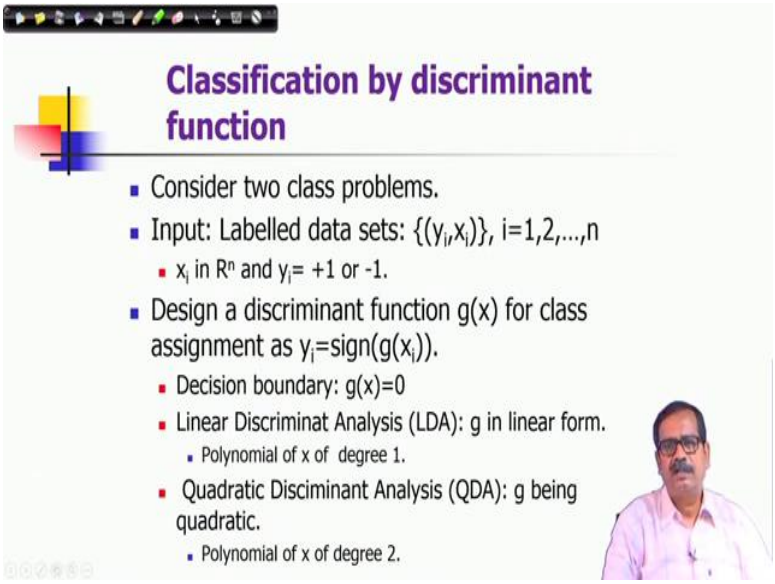
(Refer Slide Time: 08:17)



So, these are the issues, like distance measure you have to choose the most common is an Euclidean distance. And increasing K; reduces variance and increases bias which means not class prior dominates in that case. And for high dimensional space the nearest neighbor may not be closed at all. So, that is another know difficulty using it high dimensional space.

That is why not more than 20 attributes on 20 dimension 20 it is advisable to use this technique. This is just empirical observation memory based techniques. So, it means must make a pass through data points for each classification. So, these are several issues of nearest neighbors. And as I mentioned it is prohibitive for a large dataset.

(Refer Slide Time: 09:10)



So, this is the approach of nearest neighbor approach. The other approach which I will be discussing now, that is called a classification using discriminant functions. Now, in this case we should consider a two class problems to illustrate. There are extensions to multi class problems, but mostly in most cases discriminant functions are very easy to use for two class problems.

And we can extend that discussions for multi class problems. But in this particular course, we will be only considering two class problems. So, the problem statement once again let me revisit with respect to these two class problems. So, we have labeled data sets like there are n such a training samples each one as a level yi each xi.

Once again its say n dimensional feature vector; and the as I mentioned xi is a n dimensional feature vector and yi there are two classes. So, we will be using class representation by this value of yi is + 1 or - 1. Then the problem of designing classifier using a discriminant function is that this function needs to be designed.

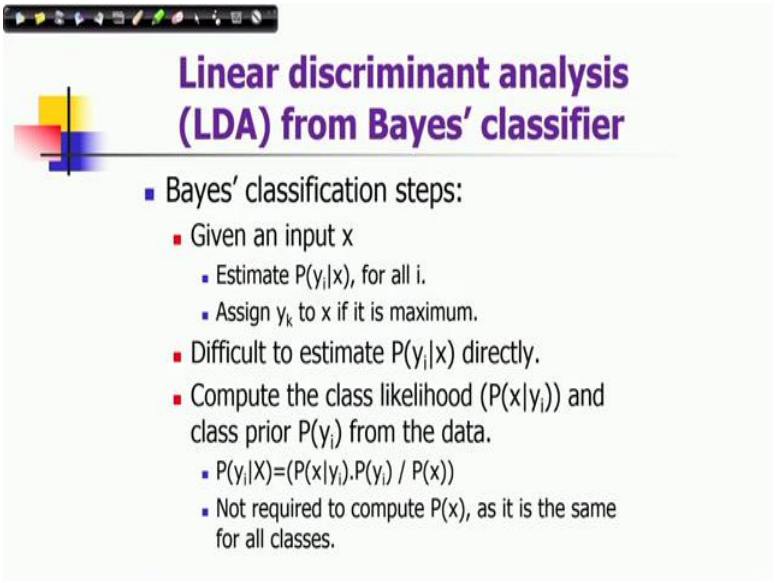
Such that, if I evaluate the function at x I then the sign up that value, which is either plus 1 or minus 1 which means positive or negative should be the class identity that is a definition of this problem. So, should note that the decision boundary is given by g(x)=0.

So, there is a boundary and which geometrically which partitions the space into two regions on one part all the values. If they put the discriminant function there it should be

+ 1 they other part would be - 1. Usually, that is the scenario of discriminant functions. And on the boundaries only and these values are 0.

When we call this analysis as linear discriminant analysis; when this function g is in linear form and it is a coordinated discriminant analysis when g being quadratic. So, the meaning of g is in linear form is that if this is the polynomial of x of degree 1. So, x is a multidimensional vector. So, for each attribute, which is taking part in the expression of functions; the polynomial degree of any attribute should not exceed 1.

(Refer Slide Time: 12:13)



So, it is a polynomial degree 1; and polynomial of x of degree two for quadratic; so, each attribute 1 second degree should not exceed 2. So, let us discuss a particular tie a linear discriminant you know function are deriving of linear discriminant functions from Bayesian classification approaches.

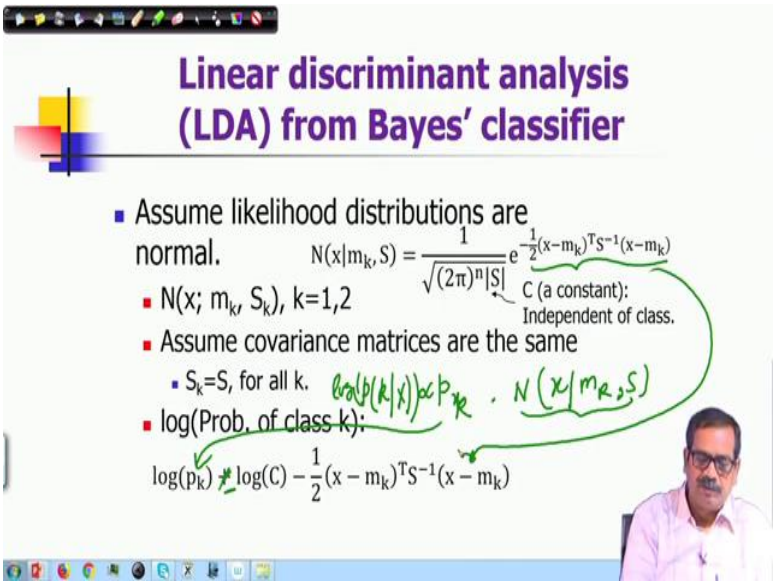
So, we call it as linear discriminant analysis from base classifier. So, the base classification steps as we have seen given an input x; we need to estimate the posterior probability P(yi|x). So, that yi in this case it could be either + 1 or - 1 there are two classes. And then assign that a value y k; that means, k th value of the k th class in to class problem it is just there are two values either y 1, y 2 two x if it is maximum.

Now as we have already discussed that it is difficult to estimate this posterior directly what we should do? We should compute the class likelihood and class prior from the data which

has been shown here. That probably of x given y i that is a class likelihood of given the i th class y i. And the class prior is probably of that i th class itself.

So, this is how posterior is related with them. And once again just to stress that we are not required to compute quality of x. And that is y only computation of those two factors are is sufficient to perform this classification.

(Refer Slide Time: 13:53)



So, now let us assume this likelihood distributions are normal. We have discussed for naive base classifier, where an unlikelihood distance distributions. We have discussed about the discrete and categorical classes. But we categorical attribute to domain actually attribute sets.

$$N(x|m_k,S) = \frac{1}{\sqrt{(2\pi)^n|S|}} e^{-\frac{(x-m_k)^T S^{-1}(x-m_k)}{2}}$$

But if I consider the continuous attribute value in a continuous domain space the attribute value occurs which we assume. We can use the parametric modeling. And let us assume that for every n dimensional endpoints diamond every n variables are in attributes. I mean it is in our combinations is that n dimensional feature vector itself follow a normal distribution.

So, here the independence of attributes search are not considered. So, this is what a normal distribution in a multidimensional space is represented to note here. There are two

parameters though the number of elements in those parameters they are not single as it was for one dimension. One is m k, which is a k th which is a 1 second and dimensional vector, which is the mean of the class it is the vector. And Sk is a covariance metrics of once again N crossing metrics but which is symmetric.

This is how the probability density functions in the multidimensional space looks when it is a Gaussian distribution. Just to simplify these expressions, we considered the denominator part as a normalizing part of this probability density function. And in this case this is a constant. We are assuming that every class has covariance metrics that is an assumption here.

So, assume covariance matrices at the same then only we can do this job. So, this is what Sk equal to S for all k. So, now, we will be expressing this particular probability by using logarithm operations. Because the advantage of logarithm is that the comparisons of values in original domain can be carried out in the log domain. Because, those values are proportional to the log values itself or log values are proportional to the origin and value itself.

And as you can see the normal distributions, one factor is an exponential exponentiation operations. If you take the logarithm operation though so, the power of that explanation operation will be basically computed. And it could make we could make it linear form or it could make it a polynomial form rather it is not always linear.

So, if I perform logarithm so, that is advantage why you should know. Apply logarithm operations over the probability values and compute the log properties or log likelihood values, whatever in there are different ways these measures are named. So, if I perform the logarithm of this expression of the posterior.

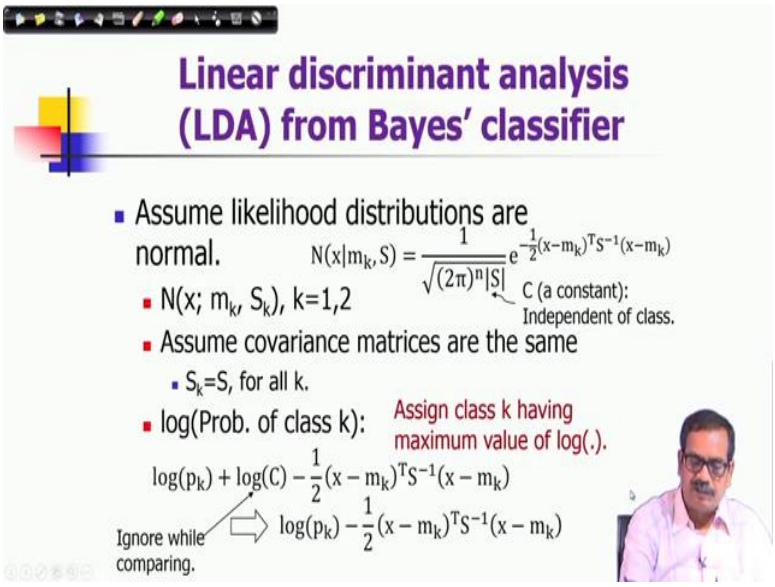
So, once you take the logarithm so, it is a locked p k and this constant. So, the posterior is proportional to p(k) that is the prior probability into. So, posterior p(k|x) is proportional to this value. Now, once I take log of this operation then log (p (k)) comes here. And from here we can see that this is a C; constraint C.

$$\log(p_k) - \log(C) - \frac{1}{2}(x - m_k)^T S^{-1}(x - m_k)$$

So, I should write here - log C; this is in the denominator, but it does not matter in our expression because we can ignore it at it is not dependent on class. It for the comparison so, because we have to find out the maximum of all log of these probabilities and then assigned that class which has the maximum value.

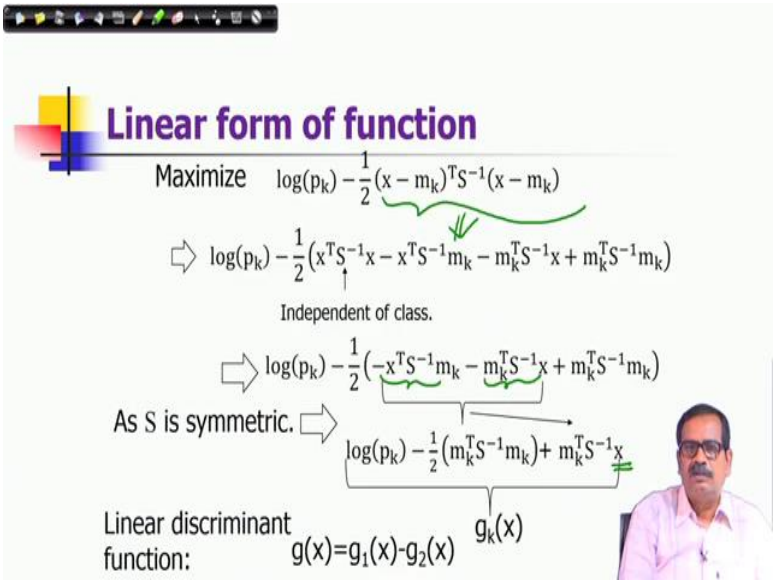
So, in the in comparing those values these constant terms will not matter. And this term is shown here that is a logarithm of exponentiation operations that would be coming there so, this is the plus.

(Refer Slide Time: 19:13)



So, as I mentioned we ignore this while comparing you should note the correction here this plus should be minus according to this representation. So, finally, it is sufficient to compare these values for finding out the class giving the maximum posterior probabilities. And we should assign that class which having the maximum value of this.

(Refer Slide Time: 19:42)



So, we will see that how actually this will give you a linear form of function. We need to maximize this value in between there some steps I will show them.

$$maximize\ \log(p_k) - \frac{1}{2}(x - m_k)^T S^{-1}(x - m_k)$$

And we can see that from that operations I have shown that is that there are two components in that operations which you need to maximize. One is the prior component that log(p (k)); other one comes from the likelihood part ignoring the constant terms.

And, if I expand them just you can perform usual algebraic operations of know multiplications using those matrices itself. So, these are the expansions of this term. So, if you expand it will be coming in this way. You can see that in my multiplications I have considered a very standard algebraic operations.

And, this part is independent of class because the covariance matrix is same for all of the class and given the data x. So, this is independent drop class once again we can ignore this for compression. And on the other hand so, that is why they in this step we are not having this particular know or term; we have only three factors. But since As S is symmetric since As S symmetric, then actually this value and these value they are same. So, simply we can add them.

So, it should be minus to slip let us suppose m k transpose is inverse x that is a value; and then if I multiply with half it would be simply this value. So, that is show the algebraic operation. So, you can see that these two terms we can combine them into a single term because they are the same you simply you can add them. And finally, we get this expression.

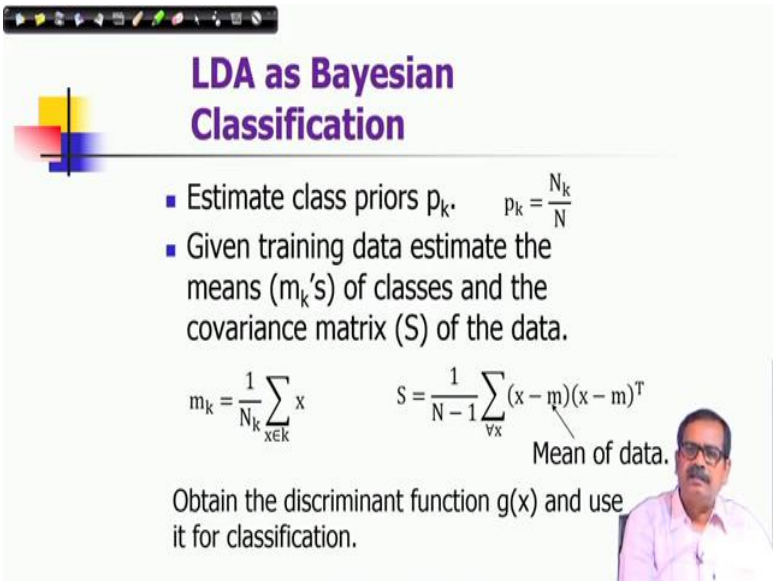
So, in this expression as a it is possible to see that there are class dependent factors these are all class dependent factors. But particularly this for discriminant functions point of view this is a variable x in this discriminant function; and which is a linear form of a discriminant function. So, this is how we can derive a linear discriminant function using a bayesian classification you know approach.

So, this is a discriminant function for these class x if I class k. And linear discriminant function can we consider as difference of these two. Because know the sign of this you have to consider a sign of g x.

$$Linear\ discriminant\ analysis\ g(x) = g_1(x) - g_2(x)$$

So, when g1(x) is maximum or greater than g2(x) the sign would be one positive. And we assign it to class 1 and when g 2 is greater the sign would be negative then assign it to class 2. So, your linear discriminant function is g x and two which is as we can see the form is a linear form.

(Refer Slide Time: 23:12)

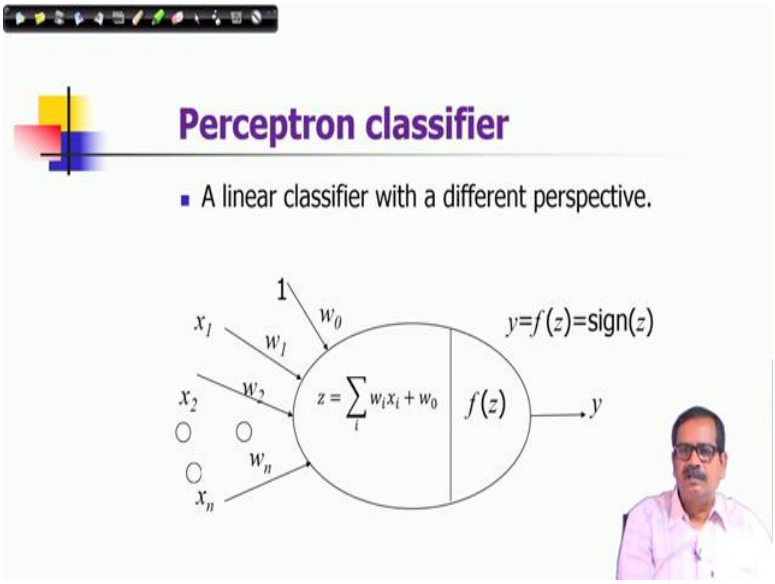


So, just to summarize this approach, that how linear discriminant function could be derived following this bayesian classification what we need to do? You need to estimate different-different class priors that is p(k). So, for estimating class priors you get the ratio of these two quantities N k is considered as the number of instances in class k and N is the total number of instances so, that is how the class prior.

And then given training data estimate the means of classes and the covariance matrix of the data in this way. That this is m k which is mean of know samples in the in the class k in this expression. And covariance can be computed from the whole data. Now, this is one know possible approach for estimating covariance there are different other refinements are there also. So, I am just presenting it for the sake of simplicity of computations.

So, you note m here is defined as a mean of the data and say it is over all mean of the data it is not the class means. So, you obtain in the discriminant function g x as we discussed and use it for classification. So, this is how you can you know perform linear discriminant analysis given the data points up to class problems.

(Refer Slide Time: 24:46)



In the next type of classification, which is also a kind of linear classifier is special kind of know classifier. Or we can see that there are relationships also and but it you can have a different perspective from this. So, let us try to understand this particular classifier which is called here perceptron, you can see there is a node and there is a computational block shown in terms of an ellipse.

So, in this computational block it has its input, where you are getting the getting a feature vector as an input described n dimensional feature vector. x 1 to x n those are the corresponding attributes in that feature vectors. And each attribute to has an weight w; I and weighted combination of this attribute values. Plus there is a constant term or which is can be considered as bias w naught. So, this is a functional form.

So, as says the function is also a linear functional form as you can see. And the class what we would you like to do the classification you have to once again its the same the sign of that. So, it is acting like a discriminant function. So, this z is acting like a discriminant function gx; and then sign op that will give you y.

So, it is nothing but the same problem, what I discussed in the in the previous example of driving example of linear discriminant analysis. So, this is also a and linear classifier. But it has been shown with this perspective it would be clear when I elaborate this aspect know later on. Let me take a break at this point and we will continue this discussion in that next lecture.

Thank you very much for your attention.

Keywords: Nearest neighbor, linear discriminant analysis, perceptron.