

Multivariate Procedures with R

Prof. Shalabh

Department of Mathematics and Statistics

IIT Kanpur

Week – 10

Lecture – 47

Classification Procedure and Analysis in R

Hello friends, welcome to the course multivariate procedure with R. So, you can recall that in the last lecture we had discussed about the linear discriminant analysis and we had talked about the linear discriminant function and the objective was that we wanted to have a linear function based on certain observations which can classify observation into one of the two population. And then we have seen that okay whether we have a base procedure and then we have extended it to a multivariate normal distribution. So, now in this lecture we are going to implement these procedures in the R software. Definitely you will agree with me that with the availability of the software many more things can be achieved which are difficult to achieve in theory. In theory I can give you the concept for example when I want to compute the probability of misclassification, sometime it may be a complicated integral but using the numerical techniques software can give us the good value on the basis of given set of data.

So, this philosophy will be extended to this linear discriminant analysis also in the R software and you will see in this lecture that first I will try to demonstrate what I have done and how I am going to achieve it and whatsoever is the outcome of the software how to interpret it. And beside that thing I will try to show you here something more. For example, if you have found a linear discriminant function how it is going to perform and when you try to adjust the capability of the derived linear discriminant function. So, one thing in this lecture may be little bit longer because my objective is that I will try to take one dataset and I will try to do each and everything on that dataset only so that you can understand it easily.

I do not want to take multiple datasets. So, that is why I want to start it and I want to finish it in the same lecture so that you can understand it easily but if you feel after

sometime that you need a break you can pause the video and can watch after sometime but I would like to complete it in a single lecture. So let us begin this lecture and try to understand how you can implement the linear discriminant analysis in the R software. So now in this lecture we are going to talk about the classification procedure, the base procedure and the procedure under multivariate normal contribution in the R software how you can do it. So, just to recall and just to refresh your memory this linear discriminant analysis was developed by Sir R.A. Fisher in 1936 to classify the subjects into one of the two clearly defined groups.

So sometime this discriminant function which you have derived is also called as Fisher's discriminant function and this analysis is also called as Fisher's discriminant analysis. So, in this case if you try to recall we had considered the two multivariate normal populations, normal μ_1 σ and normal μ_2 σ and we have assumed that an observation has equal probability and equal cost could be classified between these two populations and we have derived the classification rule here is like this. The entire region was divided into two mutually exclusive regions R_1 and R_2 and so if your σ μ_1 and μ_2 are known then you can compute this linear function and based on that you can classify the observation either into R_1 or say R_2 which are going to classify further it into as if the observation is coming from normal μ_1 σ or normal μ_2 σ . But here since this μ_1 , μ_2 and σ are known in practice so the question is how to implement this R_1 and R_2 in the real dataset.

So, what we try to do here that we try to observe two samples, right. First, we try to observe a sample x_1, x_2, \dots, x_{n_1} from the population 1 of size n_1 from normal μ_1 σ and we try to estimate here the sample mean vector that is going to give us the value of μ_1 that is the estimator of μ_1 , right. So, and similarly, we try to obtain a second sample x_1, x_2, \dots, x_{n_2} of size n_2 from normal population with the mean vector μ_2 and covariance matrix σ and then we try to estimate the population mean μ_2 as sample mean vector \bar{x}_2 , right. And then we try to obtain the estimator of the covariance matrix σ here as say here S which is here given like this. So, this is what we have obtained earlier.

So, actually, what we have done that we have individually obtained the estimate of μ_1 , μ_2 and σ and now we try to replace them back into the R_1 and R_2 region. So, the classification rule becomes here like this. So, if you try to see here this μ_1 has been replaced by \bar{x}_1 , μ_2 has been replaced by \bar{x}_2 and so on and this σ is replaced by here S . So, this is how we try to obtain the region R_1 and R_2 . So now I will try to take a sample x_1, x_2, \dots, x_{n_1} from the population 1 and x_1, \dots, x_{n_2} from the second population.

Then we try to compute their sample μ , then finally the covariance matrix. So now given an observation here x , I can compute this quantity and depending on whether it is greater than 0 or greater than 0, I can classify the observation into R1 or R2 region and that will ultimately indicate that observation is coming from normal μ_1 sigma or normal μ_2 sigma, right. So, in the linear discriminant analysis in the R software takes a data set of cases which is also known as observation as input, right. And for each case you need to have a categorical variable to define the class, right. So, you could see that I will be using a command factor to convert one of the observations which is the final observation indicating the category as a categorical variable.

So now this I would like to show you through an example how to get it done. So LDA is the command in the R software to find the linear discriminant function and to conduct the linear discriminant analysis. And this LDA command is available in the MASS package, M-A-S-S, all the upper case which is the built-in package, right. It is available in the MASS package, so you need not to download any additional software, right. One thing that you will notice that when you try to apply this linear discriminant analysis in the R software, then for a given problem the number of discriminant function will be equal to the smaller of the number of predicted variable or one less than the number of groups on the dependent variable, right.

And based on that you will have for example if you have three variables then you will be getting here two linear discriminant function. And then I will try to show you that how are you going to decide that which linear discriminant function is better to use, right. We have certain criteria. So, the command here will be like here you have to give here `lda()` or in lower case alphabets, then here is a formula or you can give the data as here x and then in the first case if you are trying to give the, use the format of formula then you have to give here data and there are many other options. And if you try to give the, use the command in the format of data say here x like this then you have to define here a grouping variable also.

So, the formula is going to be defined here like that if you try to remember the way we have done it in the multiple linear equation analysis, but it will be something like groups \tilde{x}_1 plus x_2 and so on, right. So, and in the x here, this can be a matrix data frame or a matrix containing the explanatory variable and then we will have to define here a new variable here grouping, right. So, you can choose any one of them and there are some other options here that you will see that grouping, grouping is a factor which is used to specify the classes of the observations, right. If you say that you have called say group 1, 2, 3 then I can convert it into this factor as say here say lower, middle, upper and so on.

Data is an optional data frame list or environment from which the variables specified in the formula are differentially could be taken, right.

Then you will have a prior here. The prior probabilities of class membership is indicated by this command or this option prior, P R I O R. If this is unspecified, the class properties of the training set are used and if it is present, the probability should be specified in the order of the factor levels and if you do not do anything, then it will try to take equal probability. Then you also have a here option here cv. If this is true, then it resounds the results, classes and posterior probabilities for leave 1 out of cross-validation, right.

And if the prior is estimated, the proportion in the whole indicator set are used. So that is in your control actually now and in the outcome, you will see the outcome as prior. That will indicate what are the prior probabilities that are used. You will get here the group means under the name here means and then there will be scaling that is going to be a matrix which transform the observation to discriminant function normalized so that the within group covariance matrix is spherical, right and will be the number of observations used. So now we will try to take an example and try to understand which will make the things clear to us, right.

So, I am trying to take here an example from this book univariate, bivariate and multivariate statistics using R. This is a quantity tool for data analysis and click data analysis. Data science by Daniel J. Dennis that was published by Wiley in 2020. So, I am going to consider here a data set which is on the quantitative ability, verbal ability and a training group coordinating whether individuals received say no training which is indicated by here 1, some training indicated by 2 and lot of training, mass training which is indicated by a number of 3 in learning these skills.

So, for example, if I try to take the person number 1, then the quantitative ability is indicated by here 5, the verbal ability is indicated by here 2 and then whatsoever it has achieved, right that has been classified into the group number 1, 2 and 3 in this column. For example, it is belonging to group number 1 which means no training, right. And similarly, if you try to look at here say observation number 2 that is the second subject whose score in the quantitative is here 2, score in the verbal is here 1 and then again it is classified into the group number 1 which is belonging to no training. And similarly, if you try to look at here the last value 9, this person has got score 10 in quantitative, 9 score in verbal and now this has been classified into group number 3 which is indicating the mass training, right like as here 1, 2 and here 3 like this. So now we would like to conduct linear discriminant analysis on this set of data.

So first let me try to arrange this data in a data frame. So, I try to take this data on here quantitative as here quant and which is here and then the data on here verbal as here verbal and then here the training data, this is as the train, right. So, this I have compiled in 3 data vectors and then I try to create here a data frame which is here which is stated by the command here `data.frame` and `quant` `verbal` `train`, right. So, this is my here data frame that you can see and if you try to do it on the R software, this is the screenshot, okay.

I will try to show you but at this moment it is important for you to understand what are we going to do. Now I need to create a categorical variable. Why? If you try to see here this data on training, this is here 1, 1, 1, there are here 3 categories. So, in case if you get an unknown observation, unknown person, then by looking at the scores on quantitative and verbal skill, you would like to classify the person into one of the groups, category 1, category 2 or category 3. So that is why this is a categorical variable.

So, we try to give it some name. For example, if I say group 1 will be called as suppose here none, that means no training. Group 2 will be called as say here some, that means some training has been given and group 3 will be called as say here much. That will be indicating that okay lot of training has been given. So, I try to use the command here `factor` on this data `train` and then I am trying to assign the levels here 1, 2, 3 which have been used in the data and then I try to change the levels as none, some and much.

So, none is coming from group 1, some is coming from group 2 and much is coming from group 3. Well, if you want, you can use any other name. That is your choice and then if you try to see here this `train.f` will have the levels like this none, none, none, means this is corresponding to 1, 1 and here 1. Then here some, this is corresponding to 2 and all other here much, much, much, they are corresponding to a group number here 3, right.

So, you can see here in the data set also there are 3 values for each of the category, right. So, this is what we have achieved by this command. Now we are ready to conduct our this linear discriminant analysis. So, what we try to do here? I try to upload the library `mass`, then I try to use the command here `lda()` and I try to give here the formula.

So you can see here `train.f()` is my response and `verbal` and `quant` are my explanatory variables. So I try to give it here as a `train.f` tilde `verbal` plus `quant` and it is on the data is equal to `iq.train`, right. `iq.train` is this data you can see here, this data, right. That is your

data frame. And I try to store the outcome of this command into here `lda.fit` so that I can use it again and again, right.

So can I try to call this `lda.fit` and now the outcome will start here like this. First you will see this command which has been used and this will continue further. But I can just show you that how the command will look like because I am going to divide it in different parts so that I can explain you the interpretation. The whole outcome will look like here like this, right.

That you will get on the screen, right. So, what I can do here? First, I can show you here these things so that you are confident that whatever I am explaining you that is correct and then after that I will try to take one by one all these things. So, you can see here first of all I try to create here this data frame, this `iq.train` and then I try to create here factors. I used to execute this command here factors and then you can see here that now this `train.f` is now converted from 111 to here none, none, none.

Then here 2, 2, 2, let us try to observe at a point where I am trying to click and highlight. This is converted into some, some, some and then this 3, 3, 3 at observation number 7, 8, 9 this has been converted into much, much, much and the labels are here none, some and much, right. Now after this if I try to upload this library and try to use this command `lda()`, you can see here I can clear the screen. You will get here like this. So, you can see here this is the outcome which we have obtained and this is here the same outcome which I have shown you on this screen.

Now what I will do that I will try to take different parts and we will try to explain you for example first I will try to take this part, then this part, then this part and then this part one by one. So let us try to first understand what they are trying to do because executing the command is very simple but more important point is to understand it, right. So first you try to look at this part. So, it is trying to give you the prior probabilities of the group say none, some or here much, right. So, this is what we would expect the probabilities of classification to be our group in the absence of any vectors, right.

And in the R software, this probability is set by default for our data at 0.33 that you can see here. It is all 0.33, 0.3 that 1.3, 1.3, 1.3 and this is indicating that it has used equal priors for each of the group none, some and much. And yeah well, this prior probability can also be adjusted if we have a good reason but for now, we will just use it as default. If you want to change, you can change it, alright. But my objective here is to show you how

you are you going to understand the outcome. Now I come to this another aspect which is about this here, group means, right.

So, you can see here this outcome is here like this, one of the verbal is here, then quant is here and then corresponding to none, some and much we have these different values. So, this group means they represent the mean for each level of the verbal and each level of the quant. For example, in case if I say here what is here, this 2, this 2. So, this is the mean for verbal is equal to none, this none and this value here is 2. Similarly, if you try to see here in the second row, this sum is equal to here 8 corresponding to verbal.

So this means the mean for verbal for where verbal is equal to sum is actually here 8. And similarly, if you try to see here this here 9, so this is corresponding to here much and verbal. So, the mean of verbal equal to much is here 9. And similarly, if you try to come to the second column here quant, then I can say here that when quant is equal to here none, then the mean here is 4.333333 and so on which is here like this. And similarly, when quant is equal to sum, this is here, then the mean here is 8. And when quant is equal to here much, then the mean here is 9.666 or 9.67. So, this is trying to give us the group means. Now we have here the coefficient of linear discriminant. That there is the most important part that it is now giving you two linear discriminant function because now we have here three groups.

So, as I told you in the beginning here, if you try to see here, recall here, I had told you here that for a given problem the number of discriminant function will be equal to the smaller of the number of predictive variable or one less the number of groups on the dependent variable. So now I have here three groups, say here none, sum and here much. So that is why we are getting here two linear discriminant function. It is something like, right. So now but the main thing is here that how are you going to interpret this linear discriminant function.

So, you can see here it is written here LD1 which means linear discriminant function number 1 and which is here LD2 which is here linear discriminant function number 2. So, now the question is how are you going to understand it. So, you know that this discriminant function, they are the simply function of say x_1 , x_2 , x_3 , right. So now how are you going to interpret it? So, in the LD1 if you try to see, the coefficient of verbal is 0.97 and the coefficient of quant is 0.02. So, I can say that the first discriminant function LD1 weights verbal by 0.979 and quant by 0.02983.

And similarly in the second case, in this one here LD2, the coefficient or the weight of the verbal is minus 0.59 and the coefficient or weight of quant is here 0.83, right. So, this is how we are trying to interpret it. Now I want to create that how the linear function will look like. So based on that I can write down here that LD1 is equal to 0.97 into here verbal coming from here plus 0.298 into here quant, right. And similarly, if you try to see here at LD2, I am writing here the second linear discriminant function LD2 here, say LD2 is equal to minus 0.59 which is coming from say here into verbal which is coming from here and then 0.831 which is coming from here and quant from here. So, this is how you can create this coefficient of linear discriminant and from there you can create the linear discriminant function.

Now if you want to see how are you going to take any outcome, now these are your here function. If a new observation come where the person has got some verbal scores and then some scores on the quant, right, that you can substitute here and then you can compute the value of LD1 and then you can classify the given observation into any of the categories, right. One thing you have to notice here that the coefficient which are obtained here, they are actually generated on the, they are the raw coefficient, right, and that means they have been used, they have been obtained using the raw data. If you use the standardized data, then you can obtain the standardized coefficient also and for that the command will be here that `lda.fit` means whatever is your outcome dollar scaling. Now you can recall that couple of lectures back we had talked about how to normalize the data using the function `scale`, right.

So, I will try to show you that when you are trying to use this scale data, then the linear discriminant function will come out to be different. Yeah, ideally, they should be same but it does not happen because the behaviour of the data and the nature of the data also changes, right. Well, if you, now the next question here that how are you going to understand that which of the linear discriminant function is more capable or more efficient. So, this is actually obtained by this part of this analysis which is proportion of trace and you can see here for LD1 the proportion of trace is here 0.9889 and for LD2 this trace is 0.011, right. So these two values, they can be considered as the measure of importance for each function. Yeah, how they are coming, I will try to explain you. Although it is not clear from the output because outcome is not giving all the detail but it is giving you the final value. But definitely I would like you to know that how these values have been obtained, right. But first let us try to understand what is the interpretation of these values.

So, if you get a higher value of this LD that indicates that more importance and preference can be given to the corresponding linear discriminant function. So obviously if you try to see here the LD1 has a coefficient 0.9889 and LD2 has a, the proportion of traces, trace in LD1 is 0.9889 and the proportion of trace in LD2 is 0.0111. So obviously the proportion of traces in LD1 is much much higher in comparison to LD2. So, this LD1 is going to be a favorable linear discriminant function, right. So, the relevance of each discriminant function can be computed by contrasting its eigenvalues to the sum of eigenvalues generated by the entire discriminant analysis. Well, we have not talked that because we have not talked about the mathematical part of this linear discriminant analysis.

So, this part I have not covered it. But I can share with you that it is trying to create the eigenvalues and you know that how to create the eigenvalues using the command EIGEN, right. And then based on that whatsoever are the values of those eigenvalues, they are taken and then can the relevance of each of this LD1 is obtained by finding out the eigenvalues divided by the total of eigenvalues, right. So, R does not report this eigenvalues but they are used behind the scene for obtaining the proportion of this trace output. For example, if I do try to do it manually which I have done, so for our this data the first eigenvalue turns out to be 14.35 and the second eigenvalue turns out to be 0.16. So, if you try to sum this 14.35 and 0.16, then this will give us the trace of the matrix, right.

Now if you try to see here what is the proportion of the trace accounted by the first eigenvalue here is this 14.35 divided by 14.35 plus 0.16 which is coming out to be here 0.989 and this is indicated here, right. And similarly, if you try to consider the second one that in second case this will be 0.16 divided by 14.35 plus 0.16 which is coming out to be 0.01 and this is 0.01 is indicated here in this LD2. So, this is how the proportion of traces has been computed and which gives us an idea that which of the LD1 or say LD2 is more preferable. So in this case I can say that clearly the first discriminant function is much more relevant in discriminating between groups than the second linear discriminant function, right. So now if you simply try to see here that how it will look like. So if I try to make a plot between the values of LD1 and say LD2 that means both these functions have been used and for the given data that we considered in the example for each set of data the values of LD1 and LD2 are calculated and they are plotted on say x and y axis.

So you can see here what is happening and this will give you some idea. You can see here this none is coming here very clearly. This has been classified. Similarly, if you try

to see here this here much is here separated clearly. But now if you try to look at the sum, two sums are here and one sum is here. So, it looks like something like this figure but if you try to see in comparison to none and much this figure is not very clear.

So, if you try to make here a separator here like this and this one and then there is one observation on sum which is not getting classified clearly. Now I will try to show you how you can get the same information through your analysis also. So, in order to understand that how our obtained linear discriminant functions are going to work, we do an analysis that using the obtained linear discriminant analysis function we try to use the same input data and we try to determine the category in which the linear discriminant function is going to classify the given observation. Now we also have a real data where the classification has already been done.

Now then we try to compare the outcomes. For example, in the last lecture I had taken an example that when a patient enters into the room of the doctor and in case if the person is sneezing or coughing a lot then doctors automatically classify the patient into the group of ill patients, right. So now this is what doctor is trying to do but question is in case if I try to obtain a linear discriminant function and then whether the linear discriminant function is also classifying the same patient into the patient of into the group of ill patients, if the answer is yes that means my linear discriminant analysis function and the doctor who are working on the same patient are giving the same result. But in case if the opposite happens, that doctor is trying to say that the patient is ill and obtain linear discriminant function is trying to say that the person is not ill that means there is a problem and yeah and the linear discriminant function is not working properly because we have to believe on the doctor actually more because we are trying to translate what doctor is trying to say through our mathematical procedure. So, this mathematical procedure have to be correct so that we can get the same outcome what is happening in the real life.

So, this is our objective. So, once we have obtained the discriminant function, we try to predict the group membership and, in this way, we will be able to validate the goodness of our function using the command `predict` of this `lda.fit`. `lda.fit` is the outcome that you have obtained through the command `lda()`, right. So, we first find the predicted probabilities for each of the group of the dependent variable based on the discriminant function derived and then we try to compare them. So now if you try to use this command `predict` over this `lda.fit` in the R software, then you get here outcome here like this.

So, which is here the class. Class is simply what you have used none, none, none, some, some, some, much, much, much and so on, right. And but it is trying to now indicate you

what is the class which this lda.fit is now going to give you. Do you remember that your first three observations were belonging to none, next three observations were belonging to some and remaining three observations were belonging to much. But if you try to observe here what is happening? This is now this here some is here like this. So, the 1, 2, 3 they have been classified well, some, some, some have been classified well, last two observations much and much they have been classified well but this sum is creating an issue.

So, this is the meaning of here class. Similarly, if you try to look at here this posterior, posterior is trying to give you the probabilities obtained by this LDA fit, that LDA fit is trying to classify the observation in that particular group. So, I will try to explain you and then it is giving me here the value of here x is that using this LD1 and LD2 that what are the scores when I am trying to give a particular set of data to LD1 and LD2, right. So let me try to first show you this command in the, in here this in the R software. So, if you try to see here it this command, this outcome is coming here like this.

So, this is the same outcome which I have shown you in the R console. This is here the class. You can see here first three classes are none, then next four classes are actually some and remaining two classes are actually much. And then we have here posterior probabilities and the value of here x for this LD1 and LD2, right. So let me try to come back to my slide and try to explain you what they are trying to do. So, looking at this now outcome I can just partition it into different parts. So, you can see here I have taken here three observations here in the green box and you can see that their probabilities are actually quite high, very close to actually 1.

And the remaining probabilities they are very very low that you can see here, right. So, this is indicating that there is a very high posterior probability that the observation number 1, 2 and 3 they will be classified into the group none, right. And similarly, if you try to look into the second column of here sum, then you can see here these values which I have given here under this here orange box, they are very close to here 1, right or they are quite high. In comparison to all other values, right. So, I can say here that observation number here this 4, 5, 6 and 7 there is a very high posterior probability that they can be classified to the group here sum. And if I try to come to the third column here, if you try to look at here these two probabilities in the column of much in the third column corresponding to 8 and 9, they are much much higher than all other probabilities.

For example, it is here 0.83 and this is here 0.74 and so on. So, I can say that the observation number 8 and 9, they are the posterior probability that they will be classified into the group here much or the category of much is very very high. So that is why we

will try to classify the first three observation into the group number 1 which is indicated by none. Observation number 4, 5, 6, 7 into the group or the category 2 which is classified by here or called here by sum and the remaining two observation 8 and 9 into the third category which is here much and these are here the scores that you can obtain, right. So now this is what I am trying to explain you here in very simple language. So, this posterior title indicates the posterior probabilities associated with prediction into each of the three groups on the response variable.

That what and if you try to understand, they are simply trying to say that if you try to classify this observation into this group, then this is the probability. If you try to classify the same observation into this group, then this is the probability and if you try to classify this observation into the third group, then this is the probability. So now you will try to choose the category where the probability is going to be the highest, right. So that is what we see here that for the first case, we see that the posterior probability is close to 1 for the group here none while it is much less for the other two groups, right.

That I have shown you here. You can see here or here, right. And similarly, the probability of classification into the group sum for that first case is exceedingly low, right. So likewise for the third group much also, right. So, I already have explained you. So now based on this I can see here the first three observation cases 1, 2, 3 they can be classified into the first group none. 4 to 7 cases can be classified into a second group here sum and cases number 8 and 9 last two cases can be classified into the third group much.

So, this is what I explained you. In this outcome also. So now it is clear by inspecting the classification result that the discriminant analysis did not result in a perfect prediction. Otherwise, we would have expected three cases per group. Why? Because if you try to see means originally what you had, you have here the class here none, then here sum and then here much and then originally there were three observations in each of the class but now under the predicted case you have here 1, 2, 3 that is three cases then 4, 5, 6, 7, 4 cases and 8 and 9 two cases. So, there is a mismatch in the second group. And this was exactly the same thing which was indicated here in this graphic also if you try to see that this is here trying to give some different behaviour way which I told you that I will try to explain it on the basis of numerical results also.

So now we have to understand how well did the discriminant function analysis do. So, we have a discriminant function that did their best to maximize the group discrimination because it maximized the group separation does not necessarily mean it did it well. So, once you perform this linear discriminant analysis then the next question comes here that

how to evaluate how well the classifier did its job. And we already know based on the above probabilities of the group membership that it did not do a perfect job because one observation has been misclassified. So, we would like to have a summary of all this observation that whatever LDA is doing. So, in order to do this thing, I can create here a small table and from there I would like to find out the confusion matrix.

So, confusion matrix if you try to remember this was only a compilation of the frequency value into row and columns, right. So, in order to do it I would like to go here like this that first I try to store the result of predicted LDA fit that is classes. So, this can be done by `predict lda.fit dollar class`. So, this result will be obtained separately and this is stored here as a result.

Now I would like to actually means put them in the form of a table. So, I try to use here the function `cbind` and I try to say here this result is now converted into a this like `cbind` of result. So it will come in the form of a table vertical column. And similarly, whatever is my say prior this is also I am trying to obtain from this data set `train.f` using the function `cbind`. So `cbind` inside the parenthesis `train.f` and then I try to create or join both these columns together using the command `data.frame` which is here `prior` and `result`.

So, you can see here this will come out here like this and this data has been stored in the variable `out`. So, you can see here this data is looking here like this. So whatsoever be the outcome of my `predict lda.fit` you can see here this has been classified to the class 1 and then here next 4 values into class 2 and the remaining last 2 values into class 3. Whereas in the original data `train.f` the first 3 observation were in class 1, next 3 observation were in class 2 and the remaining 3 observation were in class 3. So that is why I have artificially taken this data so that you can remember that there are 3 observations in each of the group. But now you can see here this is matching but now here there is a problem that this observation has been misclassified.

So, as I told you in the beginning that we are trying to minimize the probability of misclassification but it does not mean that the probability of misclassification will become 0 always. So that is what we want to now understand from this data that what is the report card of my LDA. So, you can see here this `train.f` column contains the actual membership of the group that is the natural group membership here. What are the real values? Then `result` contains the predicted group membership based on the previously observed probabilities which is here. Now we can note that here as we have noted that LDA got first 3 cases correct and also the following 3 cases but then classified one

observation wrongly and that is the seventh case and then it classified the last 2 observation into the correct group.

So, this observation 2 that should had been assigned to actually group number here 3 much but it was not done. It was classified into the second group. This is constituting an error in the classification or a misclassified case. So how to understand it? So, we try to create here a frequency table of this data set and this is actually now being called as confusion matrix. If you try to recall we had discussed about this confusion matrix. So, we try to obtain a summary of this how our LDA did that job and we try to compute the confusion matrix just by making a tabular representation of the data under this train.f and result.

I simply want to understand what is the frequency. Ideally there are 3 classes and then there are 3 groups. So, every group should had been given 3 observations but if you try to create this frequency table here using the command here table then train.f comma result inside the parenthesis this will give you here this type of result. So, you can see here how to interpret it. This train.f is trying to give you here this 1, 2, 3 and then there are here 3 classes here none, some and much. So, you can see here the group number 1 and here none this is coming to 3. Similarly, this 3 is corresponding to say here to some class it has been correctly classified but now on the third diagonal it is only here 2 and here this value you can see here this is 1 which is indicating the misclassification.

That means it is trying to say that this observation is coming from here you can see here this observation is coming from the group number here 3 but it is classified into group number 2. So, this is what it is trying to tell you here that this observation is coming from here this class much but it is classified into class 2. So ideally what do you expect that all the elements on this diagonal should be equal to 333 and all the off-diagonal elements should be 0. That is an ideal situation but usually in practice it may not always happen.

But this is how we are going to judge and this is here the screenshot of the thing what we have done. So, now if you try to understand what are we trying to do that of those originally in the none group on train.f all were correctly classified into the first group on the result and which correspond to also to the none you can see here like this value. So, the LDA got those first three cases correctly classified in terms of this using this linear discriminant function and for the second case here 3 now those cases originally in the some group in the data train.f all were correctly classified into the second group on the result which correspond to here some that is LDA got those three cases correct as well as in terms of classification. But now the problem is coming in the third one that this here is

here 1 and this here is 2 whereas ideally it should be actually 3 on the diagonal. So, of those originally in the much group on the data train.f only two were classified correctly and one was misclassified into the some group on the result that is the group number 2. So this is an error in the classification that we have noted through the graphical analysis also in the plot command and then in the classification analysis now in the confusion matrix right.

So, a perfect linear discriminant analysis is that the diagonal of the matrix would contain all the numbers and no numbers would appear in the off diagonal right. Hence perfect classification would have resulted into 3 across the main diagonal that means all these diagonal elements should only be actually 3 right. The same thing can also be seen in terms of a portion. So, this is very simple to create that is something like relative frequency type of thing. So, this can be created by the command here `prop.table` that is proportion of table of this confusion matrix right that we have given the name confusion.

So, you can see here same data instead of number, instead of absolute frequency it is now the relative frequency on the diagonal it is here 1 upon 3 diagonal here 1 upon 3 and then it is here like this. So, this is actually it is like a 3 upon 9 that means 3 out of 9 cases have been correctly classified, 3 out of 9 cases have been correctly classified and here only 2 out of 9 cases have been correctly classified and this is here one of the 9 cases is incorrectly classified right. So, you can see here this is how you can obtain this one right. So, this is now I try to show you these things on the R console also right.

So, let me try to copy this command so that I can save some time here right and you will see here. So, you can see here this is your here out right and now if you try to see here this is here now you are here confusion matrix right and if you try to see here what is this here value confusion it is here like this. And now if you want to create here this table in terms of proportion so you can see here it is here like this right. So, now we come to an end to this lecture I agree as I said it is going to be a little bit longer lecture but hope you will agree with me that everything was interrelated and if I try to break it then possibly it will be more difficult for you to understand right. So, now if you try to see I have shown you that how this linear discriminant analysis can be implemented in the R software and you have seen that it is not a very difficult job the main thing is that how are you going to interpret it. And when you are trying to use this software part then there are many things which are available which are difficult to compute otherwise manually.

So, but when I am trying to use here this command here LDA I have used here only very limited options but there are many other options and for those I would strongly

recommend that you try to look into the help of this LDA and there will be many options and you try to see what you can do so that you get a good outcome according to your desire according to your requirement. For that you need to take the data set and you need to practice. For example, you can take this data set itself that is a small data set very nice data set where everything is very clear. Well I have taken this data set so that I can explain you but surely these things are going to happen on a much bigger data set but definitely when I have a much bigger data set as an instructor it is difficult for me to explain you what is happening inside it but once I have explained it on a smaller data set only 9 observation which I can explain very clearly then you can believe that whatever this LDA function is doing on a bigger data set you can just believe on the outcome and then you can take a proper statistical conclusion. So, you try to practice it and I will see you in the next lecture till then goodbye. Thank you.