

Business Analytics & Data Mining Modeling Using R
Dr. Gaurav Dixit
Department of Management Studies
Indian Institute of Technology, Roorkee

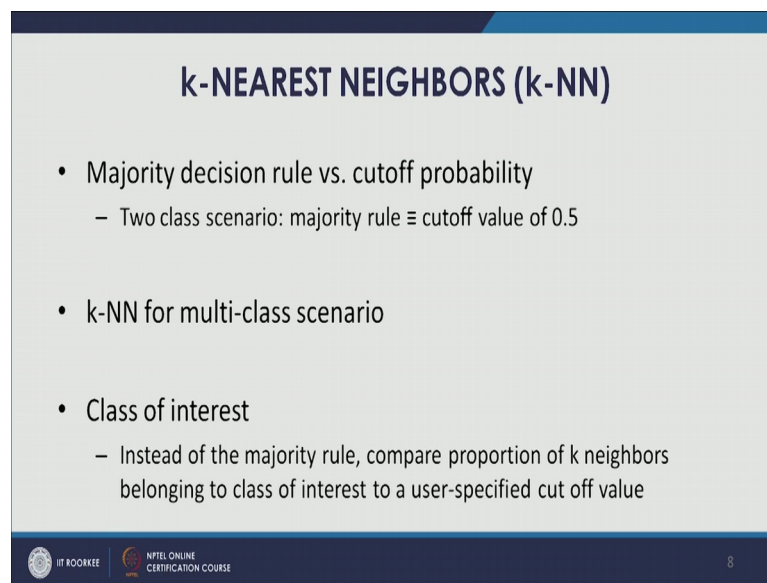
Lecture – 30
Machine Learning Technique K-Nearest Neighbors(K-NN)- Part III

Welcome to the course Business Analytics and Data Mining Modeling Using R. So, in the previous lecture we were discussing k-Nearest Neighbors, k-NN and we covered the theoretical aspect we also did an exercise, so mainly the discussion and the exercise that we had what was actually specific to the classification task right. So, let us continue the discussion on k-NN.

So, few things that we wanted to discuss with respect to the classification task. So, we will do that. So, in the example that we had done right, that was about mainly the majority decision rule. So, sometimes you might have as we have talked to in the performance metric lectures as well that sometimes we might have the class of interest and therefore, we would like to we would be interested in classifying the records to that class of interest.

So, even if it comes at the expense of misidentification or misclassification of more records of the other classes right. So, in that sense the majority rule that we have been talking about in the k-NN case right where if we find the distances that we find the k-nearest neighbors and then depending on the those nearest neighbors then we classify the new observations depending on the a majority rule right. For the classification task we find out the majority class most prevalent class among those k neighbors and then on that particular classes assigned to the new observation or classified as the class of the new observation.

(Refer Slide Time: 02:24)



k-NEAREST NEIGHBORS (k-NN)

- Majority decision rule vs. cutoff probability
 - Two class scenario: majority rule \equiv cutoff value of 0.5
- k-NN for multi-class scenario
- Class of interest
 - Instead of the majority rule, compare proportion of k neighbors belonging to class of interest to a user-specified cut off value

IIT ROORKEE | NPTEL ONLINE CERTIFICATION COURSE

Now, this majority decision rule if we connected this particular rule with the cutoff probability value that we have been using specifically in the context of class of interest or even in the when we have the equal misclassification cost or we do not have a specific class of interest and we are looking to minimize, the overall classification error, overall misclassification error even in that case also the we can connect this majority decision rule with that the probability value.

So, if we take the two class scenario, where two class scenario where we have to classify our records either to class 1 or class 0. The majority rule can is similar to the cutoff value of 0.5 and. So, for example, if we have 10 records and out of those 10 records if you know 6 records belong to class 1 and 4 records belongs to class 0. So, as per the majority rule the class 1 would actually be assigned because it is more prevalent. So, more number of records belong to the class 1 having you know 6 records belonging to class 1, it being more prevalent the class and having the majority. So, this particular class would be assigned to the new observation.

Now, we just compute the probability using the same example 10 record 6 of them belonging to class 1 and remaining 4 belonging to class 0. So, probability of a particular record belonging to class 1 would be 6 divided by 10 that is 0.6, and the probability of a particular record belonging to class 0 would be the 4 divided by 10 that is 0.4, right. So, in that case also if we follow this two class scenario we follow the cutoff value of 0.5.

So, the 0.6 that being more than this cutoff value. So, the class new observation would again be classified into class 1. So, whether we use the majority rule in especially in two class scenario whether we use the majority rule or the cutoff value concept we would get the same result. So, majority rule the majority decision rule can be easily you know connected with the cutoff probability rule that we talked about in the previous lectures.

In case we have, the same thing can be extended to m class m classes scenario, wherein if you have m classes again there also as per the majority rule we will find the most prevalent class so that class would be assigned to the new observation. So, if we talk about the cutoff value the probability value. So, for example, there are 4 5 classes they will have different probability value of you know belonging to those particular classes like 0.3 and 0.3 again 0.25, 0.2. So, these kind of if there are 3 4 classes. So, these are the kind of probability values that we might have. Again there also the highest probability value that can be used and the new observation can be assigned to that particular class of having highest probability value. So, this particular concept can be easily extended into m classes scenario. So, majority decision rule that we have been talking about can be easily connected with the probability based values and using the cutoff probability value to classify the new observations.

So, next thing that we want to talk about is the k-NN for multi class scenario. So, as we have been talking about we have been mainly discussing and the exercise that we have done that was mainly for the two class scenario right. So, k-NN all the concepts the different steps let us say look at these steps that we had discussed right. So, these are the steps, so most of these steps you can see that they can be easily extended to an m class scenario right more than two class scenario.

(Refer Slide Time: 07:16)

k-NEAREST NEIGHBORS (k-NN)

- k-NN: Finding neighbors and Classification
 - Compute the distance between the new observation and training partition records
 - Determine k nearest or closest records to the new observation
 - Find most prevalent class among k neighbors and it would be the predicted class of new observation
- Open RStudio

IIT ROORKEE NPTEL ONLINE CERTIFICATION COURSE 5

You can see the first step being compute the distance between the new observation and training partition record. So, whether it is a two class scenario or m class scenario. So, this step is going to remain same still we are going to compute the distance between the new observation and training partition records.

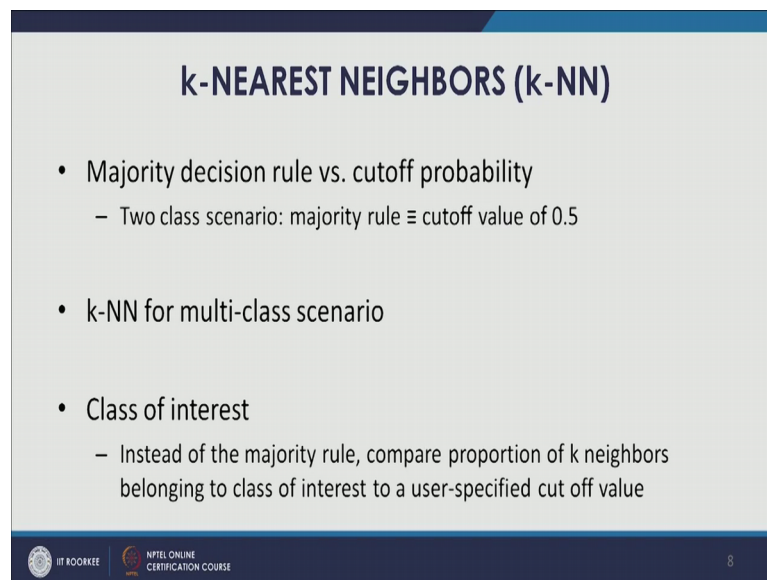
Then determine k-nearest or closest records to the new observation all right. So, that also this step will also more or less remain same the remaining k-nearest neighbor selecting that that value right, so that will also remain same now find most prevalent class among k neighbors. So, that is where things will change, but depending on whether it is a two class scenario or m class scenario. And then it would be predicted it would be the predicted class of new observations. The most prevalent class there also you can use the probability value. So, class having the highest probability value among these k neighbors. So, that can easily be assigned. So, easily this two class scenario and the exercise that we had done that can be easily extended to m class scenario.

Now, once we have talked about how that majority decision rule can be connected with the cutoff probability based method for assignment of classes now let us come to the our class of interest. So, till we have been talking about, till now we have been talking about when we do not differentiate between different classes and we would like to you know optimize for the overall error right. So, but in some situation as we have talked about in previous lecture specifically the lectures in performance matrix topic right. So, there

sometimes you might be have might have the class of interest and therefore, sometimes we might like to we might like to identify more records belonging to this particular class of interest even if it comes at the expense of misclassifying records in the other classes.

So, how do we change? How do we change our steps? For this, instead of the majority rule now we can compare proportion of k neighbors belonging to class of interest to a user specified cutoff value.

(Refer Slide Time: 09:36)



k-NEAREST NEIGHBORS (k-NN)

- Majority decision rule vs. cutoff probability
 - Two class scenario: majority rule \equiv cutoff value of 0.5
- k-NN for multi-class scenario
- Class of interest
 - Instead of the majority rule, compare proportion of k neighbors belonging to class of interest to a user-specified cut off value

IT ROORKEE | NPTEL ONLINE CERTIFICATION COURSE | 8

So, now we will not be looking at the probability value for all the classes among k neighbors and then finding out the class having the highest probability value. So, instead of that instead of following this majority rule or the respective probability base method cutoff probability method will focus on the class of interest and we will try to find out the proportion of k neighbors right, proportion of k neighbors that belong to the class of interest; that means, also in the probability terms the probability of probability of a particular record belonging to the class of interest among those k neighbors. So, once we compute that probability then we can compare it to the user specified cutoff value because we are more interested in finding out the class of interest members. So, therefore, instead of having a higher you know in a two class scenario we would like to have the 0.5 is the cutoff value.

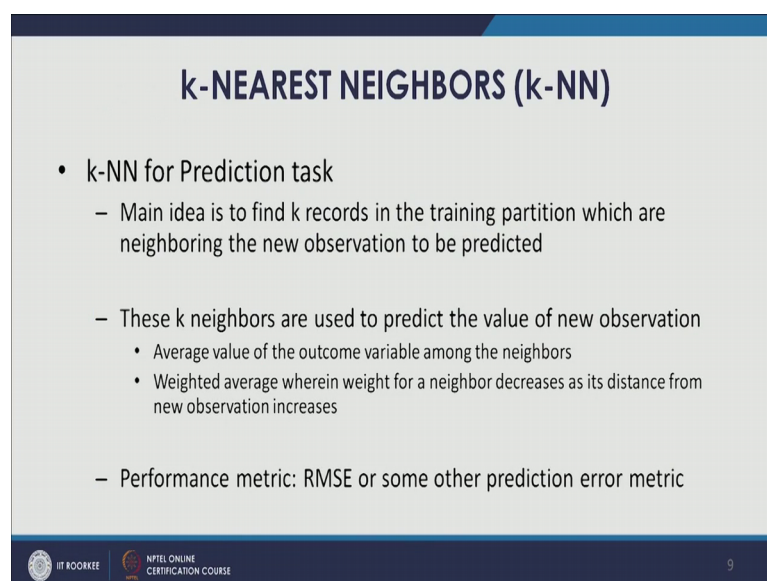
Even if for the class of interest also if it is a two class scenario we would like we might specify a lower cutoff value than 0.5, it could be 0.4, would be 0.3, 0.2 depending on the

scenario. So, if it is the class of interest is quite rare in the data set that we have then probably we might lower it down to 0.2 or 0.1 all right. So, depending on the situation we might lower down the user specified this cutoff value from 0.5 to as low as 0.1 all right.

So, now, the proportion of this probability value among k neighbors the probability value of belonging to class of interest that would be compared to this a specified a cutoff value. So, if it is greater than this value cutoff value specified cutoff value then the class would be that particular record observation would be classified to the class of interest otherwise not. So, we are not interested in classifying the records into other classes. So, our main focus is the class of interest. So, we just focus on the, we are just compute the relevant probability value for that class of interest and then we compared it to the specified cutoff value. So, eventually we would be able to identify more of more records belonging to class 1 and then there are going to be generally typically there are going to be more misclassification for other classes.

So, let us move forward. So, till now the discussion that we had about k -NN modeling this was with respect to classification tasks can be used k -NN for prediction tasks. So, yes it can be used.

(Refer Slide Time: 12:38)



k-NEAREST NEIGHBORS (k-NN)

- k-NN for Prediction task
 - Main idea is to find k records in the training partition which are neighboring the new observation to be predicted
 - These k neighbors are used to predict the value of new observation
 - Average value of the outcome variable among the neighbors
 - Weighted average wherein weight for a neighbor decreases as its distance from new observation increases
 - Performance metric: RMSE or some other prediction error metric

IT ROOKIEE | NPTEL ONLINE CERTIFICATION COURSE | 9

So, as you have seen that in the classification tasks the outcome variable that is the categorical that has to be categorical variable and if it is a numerical variable then we

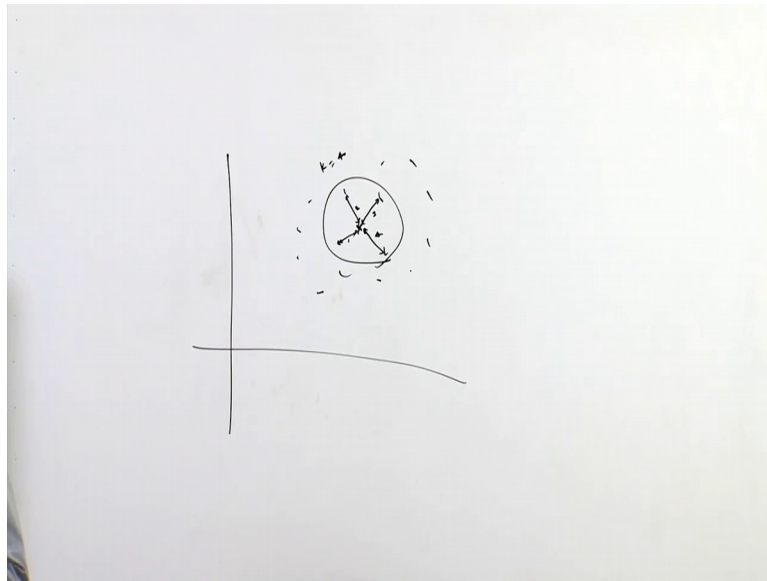
will have to convert it into a categorical variable through winning. So, for a classification task we will have to do that. But for prediction tasks our variable has to be a numeric variable or continuous variable. So, let us discuss how k-NN is different for a prediction task in with respect to in comparison to classification task.

So, again main idea is to find k records in the training partition which are neighboring the new observation to be predicted. So, this particular this particular step does not change right. So, main idea is to find k records in the training partition which are neighboring the new observation to be predicted. So, this is step is remains as is.

Now, let us look at the second step. So, these k neighbors that we have identified in the first step are used to predict the value of new observation. So, earlier be used to predict the class of new observation in the classification task. Now, we want to predict the value of new observation now how do we do that. So, in the classification task we had this majority rule decision or the you know higher probability value you know class having the highest probability value. So, that could be used.

Now, instead of the majority most prevalent class now because this is a prediction task we would be taking the average value of the outcome variable among the neighbors. So, the k neighbors that we have identified in the previous system right. So, for all those variables we take the average value of the outcome variable and that and this particular value would actually be the predicted value for the new observation. Sometimes researchers or analyst they might prefer the weighted average value and generally this weighted averages is computed in a fashion that weight for a neighbor decreases at its distance from the new observation to be classified increases.

(Refer Slide Time: 15:13)



So, the points if we want to say this, different points could be their belonging to different classes for our class of interest right. So, the weights for the points for example, these are the close y points. So, we are having the k value of 4. So, therefore, the weights this point seems to be this particular neighbor seems to be closer to this new observation all right, this is let us say this is the distance and based on the distance and value of k we have identified 4 nearest neighbors. Now, we try to compare these neighbors probably this is the smallest, and then followed by this one, then let us say this one and then this one.

So, as the distance increases right as the distance increases the weight would be decreasing. So, we can give more preference to the record which is closer to the new closer to the new observation. So, more weight for this particular record followed by this record than this and this. So, as the distance between records between the record and between the new observation and the record increases the weight will actually decrease. So, weighted average is also sometimes used in k -NN for prediction tasks.

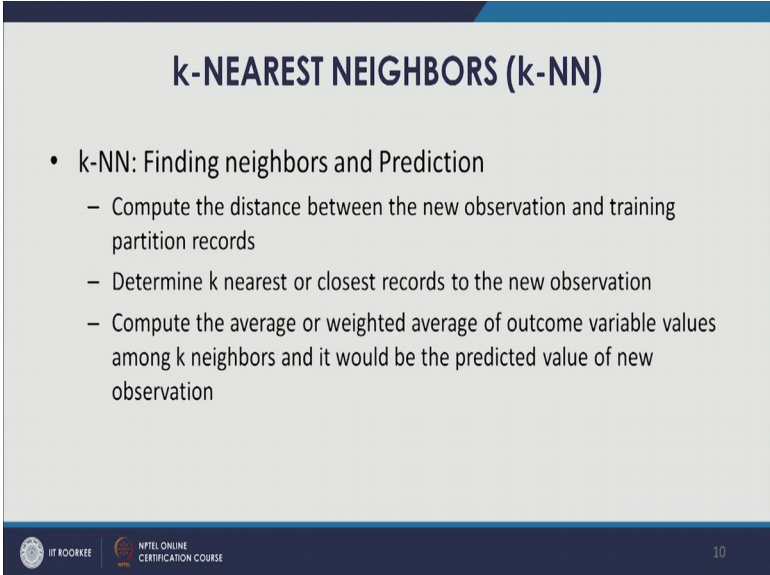
Now, the change in performance metrics. So, earlier we were looking to minimize the overall error overall misclassification error right. So, that was the metric for classification tasks. When we talk about the performance metric has discussed in the performance metrics topic different lectures that we had right. So, we talked about various metrics that could be used for the prediction task. So, we also talked about THE

R M S E. So, in this case R M S E is generally used, but we can also use some other prediction error metric that we talked about in that in those lectures.

So, if we compare the steps of k-NN for classification task and the prediction task, the first step remains same that we try to identify the k records right the k-nearest neighbors and then either we pick for the for the classification we find out the most prevalent class. And for the prediction we try to take the average value as the predicted value among the k-nearest records.

So, if we are interested in, we are interested in understanding the steps in more detail, finding neighbors and prediction.

(Refer Slide Time: 18:07)



k-NEAREST NEIGHBORS (k-NN)

- k-NN: Finding neighbors and Prediction
 - Compute the distance between the new observation and training partition records
 - Determine k nearest or closest records to the new observation
 - Compute the average or weighted average of outcome variable values among k neighbors and it would be the predicted value of new observation

IIT ROORKEE | NPTEL ONLINE CERTIFICATION COURSE 10

So, let us go through these steps once again. So, number one is going to be compute the distance between the new observation and training partition records right.

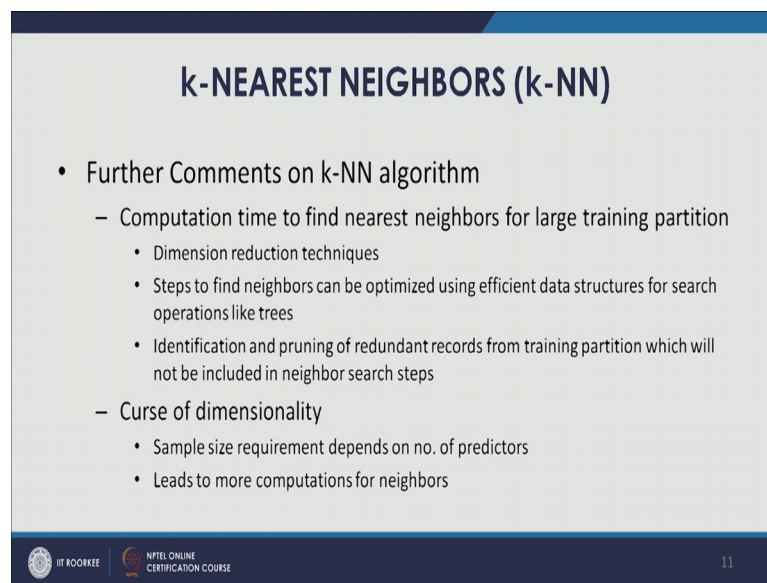
So, we can we will have to compute this compute the distance between the new observation and training partition records. So, if there are more records in the training partition then we will have to compute more such distances right even for just one observation one new observation, if there are more records in the training partition more distance computation would have to be performed. Once this is done determined k-nearest or closest records to the new observation which is now easy to do, we can just sort those records in the you know in the increasing order and first few records which are

closer first k records which are the closest or nearest having the smallest distances and they can be picked as the k -nearest or closest records to the new observation. Once this is done then we can compute the average or weighted average of outcome variable values among k neighbors as we talked about and now this particular value this average value or the weighted average value of outcome variable values of the among these k neighbours that can be that is going to be the predicted value of new observation.

So, let us talk about a few more specific points on k -NN algorithm. So, some of the some of the advantages of k -NN are very obvious. So, for example, simplicity, you can as you might have understood through these steps very simple steps that we have been discussing. So, simplicity is the advantage and k -NN algorithm and it is a nonparametric approach so we do not have to estimate any parameter which is from any assumed functional form for example, as we talked about in linear regression multiple linear regression and there we have to estimate the betas and other parameters right. So, we do not have to follow any such you know a functional form, linear or other forms and we just have to compute, we just have to measure the similarity and that to you know we have lots of distance metric which can be used as distance based similarity matrix and that are generally used, simple and nonparametric approach. So, these are some of the advantages of k -NN.

And as we have been, as we have talked about if we are dealing with a very large data set then in that case we slide the value of k to a lower value than this computation problem that can also be handled and large number of large number of you know observation you know that will not you know that can also be classified or predicted.

(Refer Slide Time: 21:24)



k-NEAREST NEIGHBORS (k-NN)

- Further Comments on k-NN algorithm
 - Computation time to find nearest neighbors for large training partition
 - Dimension reduction techniques
 - Steps to find neighbors can be optimized using efficient data structures for search operations like trees
 - Identification and pruning of redundant records from training partition which will not be included in neighbor search steps
 - Curse of dimensionality
 - Sample size requirement depends on no. of predictors
 - Leads to more computations for neighbors

IIIT ROORKEE | NPTEL ONLINE CERTIFICATION COURSE 11

Now, let us look at some of the problems that we might have to encounter in k-NN modelling. So, first one is the computation time to find nearest neighbors for a large training partition. So, of course, if the value of k is the optimal value of k is on the higher side in that case we will have to do a lot more computations. So, computation time to find nearest neighbors for a larger training partition. So, because there would be a more number of observations, more number of records, because for a new observation we have to compute distances with each of the records that are present in the training partition. So, if the partition is quite large then a lot more computation will have to be performed and, so this would increase the computation time.

Now, dimension reduction techniques. So, those can be applied to you know to handle this problem to manage this problem. So, we are able to reduce the number of variables the set of predictors that are going to be used for k-NN modelling, then definitely this particular problem this particular issue can be managed can be handled. We will have less number of computation to perform as you know that we use the Euclidean distance metric to compute the distances, and there if we have less number of coordinates right, less number of coordinates you know to less a fewer number of computation would have to be performed our Euclidean distance formula would be smaller and having fewer number of computations. So, dimension reduction techniques would definitely help. So, we should have just the most useful predictors in our model and that would also reduce the computation time for k-NN.

Now, another approach to handle this problem could be steps to find neighbors can be optimized using efficient data structures for such operations likely used. When we want to find out the k-nearest neighbor right, if the value of k is on the higher side right, it can take a lot of computation time search operation, search algorithm, would have to be can actually there are many optimized more efficient search algorithms. So, that would be used. Some data structure efficient data structure for example, trees are also available which can significantly reduce this particular time, the search time right. So, those can be used. So, that the steps to find neighbors they can be optimized. So, in that sense also we would be able to reduce the computation time to find nearest neighbors.

Now, the another thing that can actually be done is the identification and pruning of redundant records from training partition and now these records which will not be included in neighbor search steps. So, there are going to be a few records depending on the data set if the data set is quite large there are going to be many records which might not be many records in the training partition which might not be required for the neighbor search steps right. So, they will always be crowded by other records which also you know fall in the same class especially if it is a classification task. So, many many such records would also for they will be carried by the record belonging to the same class. So, therefore, you know we can avoid searching through those records or even computing you know distances for some of those records because if they are not going to figure out in that k neighborhoods or the search steps. So, some of those records can be identified and pruned. So, that would reduce the number of computations right. So, this can also be done we can identify and prune the redundant records. So, that when we do our neighbor search steps when we perform our neighbor search steps. So, we want we would be able to avoid we will be able to reduce some of the computations, some of the required computations.

Now, the next problem that can be associated with the k-NN algorithm is the because of the curse of dimensionality. So, we have been talking about if the there are more number of predictors. So, another problem might figure in specifically in the k-NN context because if there are more number of predictors as is as written in the first point under curse of dimensionality this sample size the sample size requirement depends on number of predictors. So, if we have more number of predictors the number of observation that we might require in our data set to have a useful model that would also increase. So, we

have more number of predictors, he would be requiring more number of cases, more number of observation and our records. So, that would also increase the number of computations in a k-NN algorithm because the in the training partition as well we will have a more number of records and therefore, for a new observation we will have to compute the distances and then you know search for the k neighbors. So, those computation would also increase.

So, in the previous the previous point we talked about dimension reduction techniques so that also if we do not handle that so probably this is one of the thing that should be done dimension reduction techniques would be applied to have the useful set operators only otherwise other problems could also come in one related to sample size, the number of observation that requirement would also be on the higher side if we include more number of predictors. So, as we discussed if we do not handle this curse of dimensionality this sample size the number of predictors more number of predictors therefore, the bigger sample size and therefore, more number of observation and that would eventually lead to more computations for neighbors whether it is for distance computation that is for the distance computation to identify a nearest neighbor or then whether it is for selecting the k-nearest neighbors.

So, even with these limitations the k-NN algorithm in some situations it outperforms so many other algorithms because of this you know depending on the requirement the optimum value of k could be lower down, and still will have useful results. And in some situation this still remains one of the useful algorithm.

So, with this we conclude our discussion on k-nearest neighbour. In the next lecture this will start our discussion on naive bayes.

Thank you.