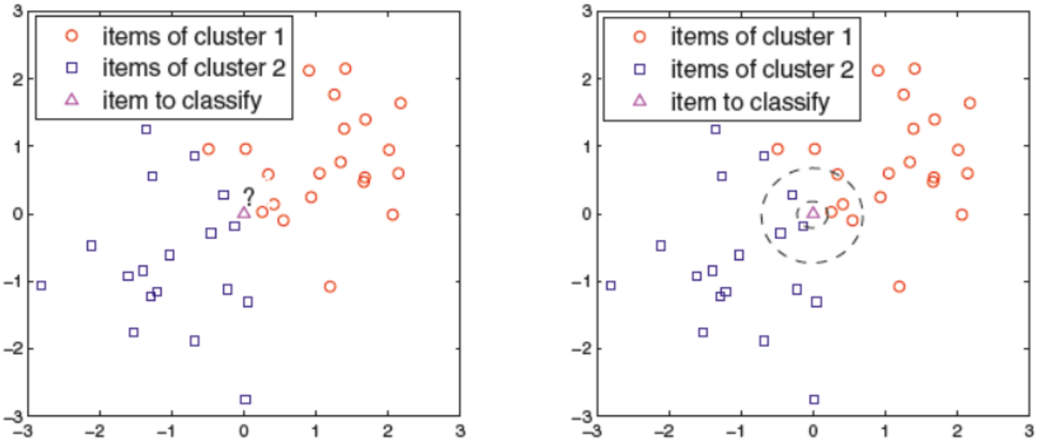
Course Name - Recommender Systems
Professor Name - Prof. Mamata Jenamani
Department Name - Industrial and Systems Engineering
Institute Name - Indian Institute of Technology Kharagpur
Week - 02
Lecture - 09

Lecture 09: Introduction to machine learning-II

I welcome you all once again to our lecture on Introduction to Machine Learning part 2. This is the 9th lecture of this series. So, here we will be talking about supervised approach again, but now we are talking about the classification. As I have told you in case of regression which is also a supervised approach, we did not discuss everything about regression or its extensions to basis function and neural network. Rather we tried understanding that why to fit a model, how to make it non-linear, when we try minimizing the error, why to take care of the over fitting problem and how to take care of the over fitting problem considering L 2 and L 1 norm. So, we also discussed about the model flexibility and interpretability, but in all this one thing was common that we were trying to predict some continuous variable.

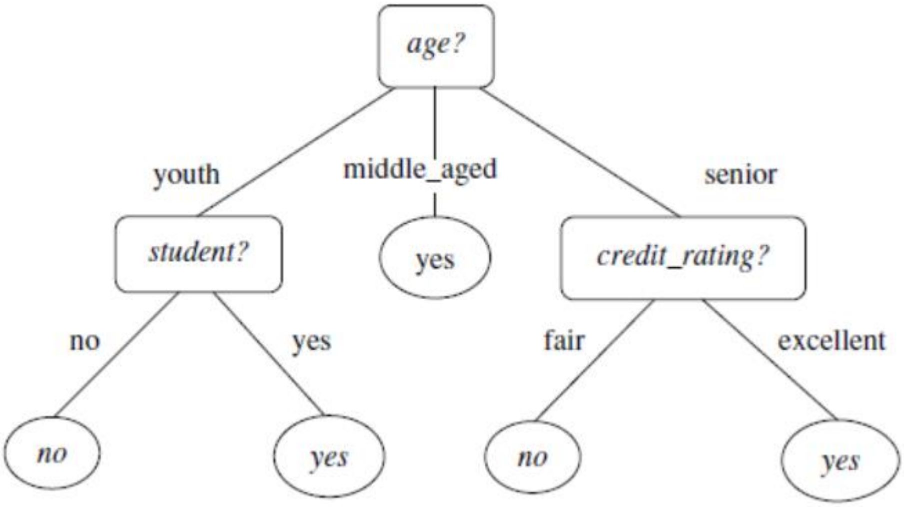
In case of classification problem, a classifier is again a function that time also we were trying to discover a function, but here also we are trying to discover a function which will map the feature space to a label space. Label space in the sense now our output variable is a qualitative variable which is represented of course, in a quantitative term, but it is basically a nominal or ordinal variable. So, for example, we may be determining whether the rating is 1 or 0 in that case it is a binary classification problem, whether a person will buy an item or not binary classification. Similarly, the rating can be from 1 to 5.



So, in case it is a it is again some kind of categorical variable and they are ordinal in scale. So, first one in this category is the KNN classifier, K nearest neighbor classifier. So, this K nearest neighbor classifiers are the very simple machine learning algorithms and quite widely discussed. So, here this case the model is not build explicitly. Every time a new observation comes with respect to the labeled data set it is it has to be found out whether it is closed to the first category or to the second category.

So, therefore, we every time we have to compute the neighborhood as when a new observation comes in. So, here the idea is if the record falls in a particular neighborhood where the class level is predominant it is because the record is likely to belong to that very same class. So, the approach is you find out the K closest points nearest neighbors from the training record and assign a class level according to the class level of the nearest neighbor. So, if in your vicinity you have 10 points and out of that 7 points are positive class and remaining points are negative because out of 10 points 3 points are negative class. So, it belongs to the positive class.

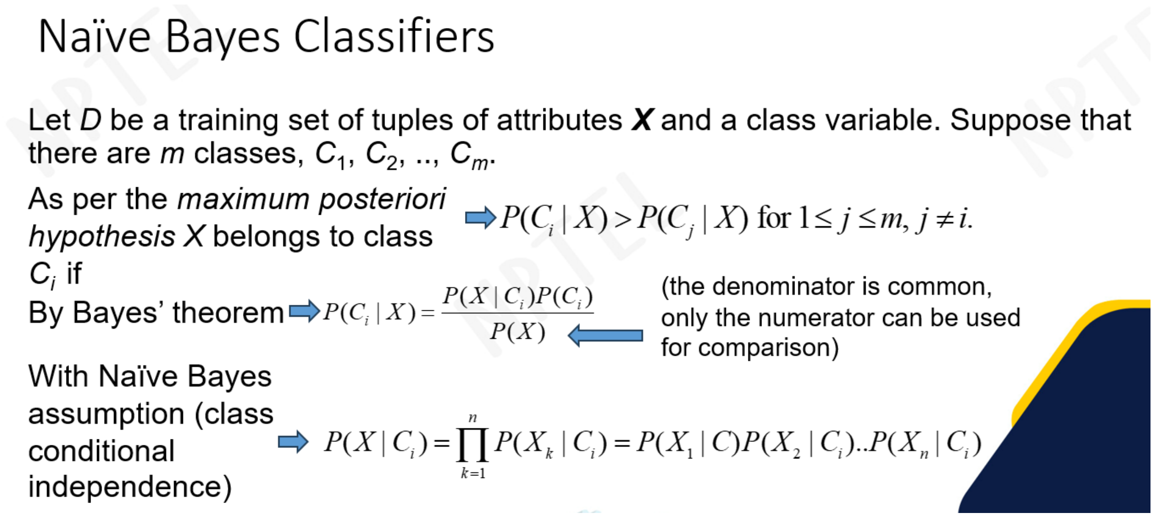
So, to this is one pictorial representation of what I said, you make the neighborhood in this neighborhood this is the point whose which is to be classified this is the point which is to be classified. Now, if you look at the neighborhood exactly 3 3 are there. So, now, increase the neighborhood little bit increase the neighborhood little bit, now 4 points of this cluster 1 type. So, which means you will classify this to be belong to the red cluster. Now, the most challenging issue in case of KNN is how to choose the value of K if K is too small the classifier will be sensitive to noise points and if K is too large the neighborhood might include too many points and decision making will be difficult.



Next category is your decision tree. The decision trees are classifiers on a target attribute in the form of a tree structure. Here it has 2 kinds of node the decision nodes in which the nodes in this nodes a single attribute value is tested to determine to which branch the sub tree will which branch of the sub tree it applies. Then in case of there is second category of nodes which are called leaf nodes that indicates the value of the target attribute. There are many algorithms to construct this. So, these are few Hund's algorithm CART, ID3 and so on. So, this is one example of a decision tree. Here at as I told you there are 2 category of nodes 2 category of nodes the decision nodes and leaf nodes. So, these are your decision nodes this is decision and these are leafs. So, decision nodes and leaf nodes make this tree.

So, while making these decision nodes some attribute selection measure has to be used. So, this attribute selection measure what it does? It splits the data set that best separates the given that best separates the data into 2 non overlapping partitions. So, this data set D has to be now partitioned into 2 parts. Now, for the first partition the next decision node is at this side of the tree on the right side of the tree. For the second partition this may be this is partition 1, this is partition 2.

So, the data which was D here made 2 partitions. So, this from this partition now this becomes the decision node for this partition this becomes the decision node. And you continue till you reach at a point where you have all the leaf nodes I mean all the data records has the same label variable. Let us say after this point you do not have to further partition the data point. So, there are 3 popular attribute selection measures information gain, gain ratio and Gini index.



Naïve Bayes Classifiers

Let *D* be a training set of tuples of attributes ***X*** and a class variable. Suppose that there are *m* classes, $C_1, C_2, .., C_m$.

As per the *maximum posteriori hypothesis X* belongs to class C_i if ⇨ $P(C_i \mid X) > P(C_j \mid X)$ for $1 \le j \le m, j \ne i.$

By Bayes' theorem ⇨ $P(C_i \mid X) = \frac{P(X \mid C_i)P(C_i)}{P(X)}$ ⇦ (the denominator is common, only the numerator can be used for comparison)

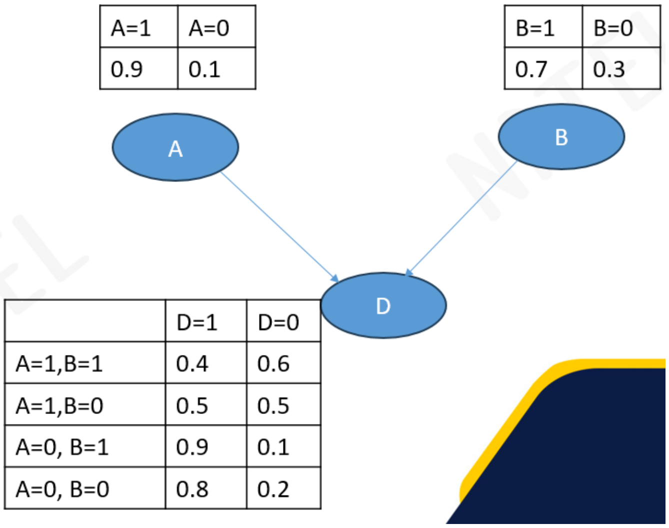
With Naïve Bayes assumption (class conditional independence) ⇨ $P(X \mid C_i) = \prod_{k=1}^{n} P(X_k \mid C_i) = P(X_1 \mid C)P(X_2 \mid C_i)..P(X_n \mid C_i)$

As I told you we are not going to discuss in detail. Now, during and mostly the during content based recommender systems will be using this such kind of models. So, when we discuss about those models in detail that that time when we use this models that time we will be discussing this in detail. So, next category of classifiers are your Bayesian classifiers. The Bayesian classifier classifiers are provides a probabilistic framework for solving classification problem.

So, it is based on the definition of conditional probability and Bayes theorem. The concepts of priors is very important as they represent our expectation or prior knowledge which tells us about what is the true relationship among the among the variables. So, this prior knowledge has to be extracted from the existing data and using this certain posterior probability has to be found out. So, there are 2 widely approaches for this the first one is your Naive Bayes classifier. In case of Naive Bayes classifier what happens? If you have a data set D containing a set of tuples with attribute x and a class variable class variable.

Then if we have m number of such classes it can be binary as well m number of such classes. Then given a new pattern when a new pattern comes what exactly has to be found out? Given the attribute x find out which class it belongs. So, therefore, you take something called a maximum posteriori hypothesis. So, this tells to which class to which class x will belong. So, x will belong to class C if this condition is satisfied.

That is the conditional probability of C i given x is higher than the conditional probability of any other class given x. So, now, our aim is to find out given a new observation x we have to find out all these C i all these conditional probabilities C i given x. So, once we find it out whichever is the highest the data will belong to that particular class. Now, coming to the Bayes theorem from Bayes theorem we know that this conditional probability can be determined from this formula. So, which means for each of the C i we are supposed to use this formula.



A=1	A=0
0.9	0.1

B=1	B=0
0.7	0.3

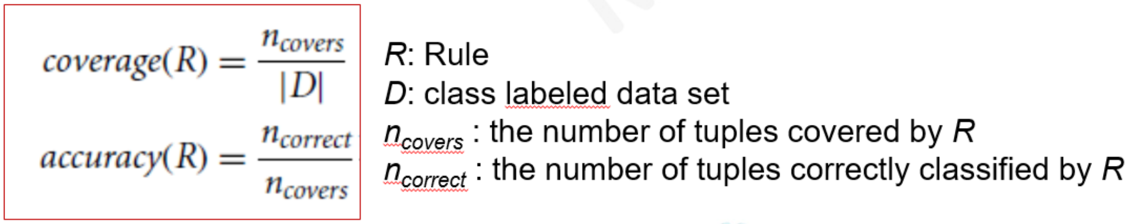
	D=1	D=0
A=1,B=1	0.4	0.6
A=1,B=0	0.5	0.5
A=0, B=1	0.9	0.1
A=0, B=0	0.8	0.2

So, when we use this formula for all then this P x becomes common. So, therefore, ultimately we need to know this prior probability values that probability of occurrence of C i from the data and probability of x given C i. So, if we found if we find and keep this values with us then whenever a new observation come we can find out this values. Now, again to make the life further simple we assume that something called Naive Bayes assumption which is called class conditional independence. So, which means it assumes that there is no dependence among all the attribute values.

So, which means when because they are independent we to find out this values of x given x i we can find out x i given C i then x 2 given C i x n given C i. So, all these values we can find out individually from the data and multiply to get this. In case of Bayesian network which also uses this the Bayes rule we now connect the data points in terms of a directed acyclic graph. So, basically it has 2 components one is a directed acyclic graph this with now in this particular example a b and d are 3 variables and this is the directed acyclic graph. Now, with each node you have a conditional probability table.

So, here d depends on both a and b. So, a has can take 2 values 1 and 0 with this probability and b can take value 1 and 0 with this is the probability. So, which means when whenever d is taking 1 d is dependent on a and b. So, it a and b because both of this is one example in which we are considering the values of this individual variables are binary, but this may not be true this can they can be any number of categories as long as we make this conditional probability tables. So, we have these combinations 1 1 1 0 0 1 1 1 0 0 1 and 0 0 and we already have this conditional probability values.

So, whenever a new data point comes we will be able to determine what is the probability. So, one and when we use it for decision making one of these nodes has to represent as the as the as the as the as the as the as the as the because now see what exactly we are given a data record x 1 to x p we are supposed to determine y. So, all this p number of variables and y all of them will be connected to each other through a network. And we are supposed to determine the value of y depending on how what is the interconnectivity among all this and by simply multiplying in this multiplying the corresponding conditional probability within this network we will be able to generate this value. The second one the next one is the rule based classifier.



$$coverage(R) = \frac{n_{covers}}{|D|}$$

$$accuracy(R) = \frac{n_{correct}}{n_{covers}}$$

R: Rule

D: class labeled data set

n_{covers} : the number of tuples covered by *R*

$n_{correct}$: the number of tuples correctly classified by *R*

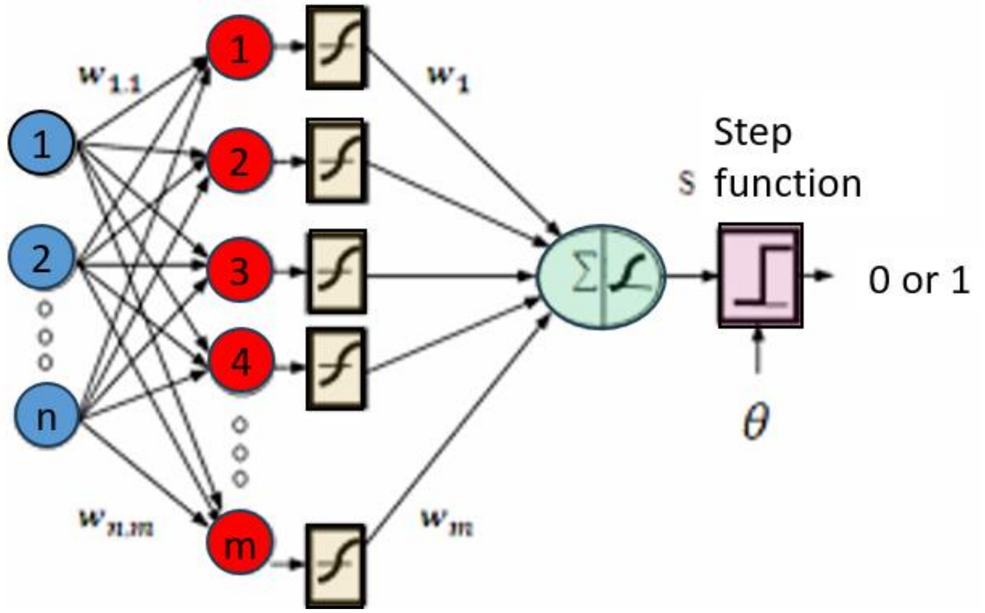
In case of rule based classifier so far how many classifiers we have studied? We studied about tree based classifier, then we studied about Bayesian classifier where we studied about Naive Bayes classifier as well as Bayesian belief network. Now, there is another category called rule based classifier. In fact, these rules can be generated from both decision trees as well as Naive Bayes. However, there are other ways to generate rules as well, but whatever may be the case the in case of rule based classifier we have to have a collection of rules to decide the class. So, these rules have two things one is antecedent another is consequent.

For example, in this rule this is the antecedent and this is the consequence. So, this rule antecedent or condition is an expression made of attitude conjunctions and the rule consequence is a positive or negative classification. Now, a rule can be assessed by its coverage and accuracy. Typically from a whatever process you follow whether it is decision tree, whether it is Naive Bayes you will be generating large number of rules. So, keeping all those large number of rules has certain computational implication as well.

So, whenever you determine the class you have to go through all the rules that is available to you. So, instead of that you can assess the rules and keep few rules with you. So, what is this assessment criteria? There are two widely adopted assessment criteria coverage and accuracy. This coverage defined by n cover by the number of number of elements in the label data set and accuracy is defined by n correct by n cover. Whereas, this n cover is the number of topples covered by the rule R and n correct is the number of topples correctly classified by R in the training data set.

This artificial neural network which in case of regression we studied as mechanism to connect the input variables to some numerical variable this can also be used as a classifier. So, how do you make it a classifier? You make certain additional step functions some kind of some kind of threshold function that can indicate the class. If the value is above a certain threshold you say it is becomes let us say class 1 and below it becomes class 0. And this can also be extended to multi class classification problem where you will have let us say you have 5 classes. So, you will have 5 output node and depending on whichever output node gives you the highest higher value you say that the item belongs to that class.

And it has to be trained just like I we discussed at the while discussing about the supervised learning. Again some error function has to be minimized and typically here the back propagation algorithm will be used or certain improvement over those back propagation. But error has to you have to compare the while training you have to compare from the available pattern you have to compare the output category with that of actual the actual output category with that of predicted output category and you have to now iterate. This is a classifier which is different from all other classifier. How it is different from all other classifier? Because here in case of SVM this support vector machine in this case you have to remember some of the training patterns.

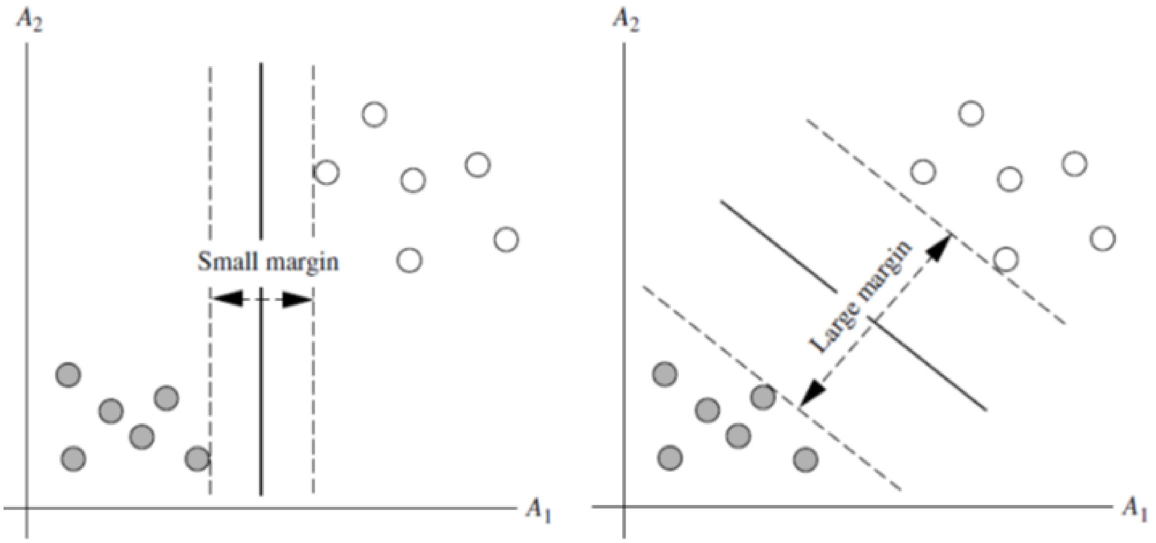


How we do it let us try to see. Now, this goal of the support vector machine classifier is to find out the linear hyper plane that separates to data in such a way that the margin is maximized. So, this is a case in here also you are separating here also you are separating both the pattern same pattern. But here in this context this margin of separation is much more than this higher it is better separated both classes are. So, the idea is to find out the idea is to maximize this distance to find out this is this is your optimally separating hyper plane. This is the optimally separating hyper hyper plane and this is your margin of separation.

So, you have to find out this hyper plane. So, that this margin margin of separation is maximized. So, you have to make some corresponding optimization problem and solve it. But while doing so, you try pushing this boundaries so much that some of the points will actually lie on this boundary on both the sides. And you try as much as possible that this as many as less number of points will lie on this boundary. And this there is in the formulation you can make certain flexible flexibility to make this choice like how to make it little bit this margin little bit flexible and decide how many such support vector you would like how many such points on the boundary you would like to keep.

So, these points which are there on the boundary or near the boundary are called support vectors. So, these support vectors as I told you while you predict when a new predict pattern comes you have to suppose to predict the class whether it belongs to this class or whether it belongs to this class. So, you are suppose to so, you are suppose to predict the class. So, in case while you try separating this margin of separation some of the points will lie on this on these boundaries.

So, those points are called support vectors. Now, your aim is to provide certain parameters so that this margin of separation can be made flexible and you can decide how much to go and how many support vectors to keep. So, once you decide this support vectors and the model parameters when a new pattern comes unlike in other classifiers where you forget all the telling patterns after the model parameters are learned here along with the model parameters you are suppose to also remember this support vectors. Now, it may so happen that while deciding this margin of separation some of the there is I mean there are this is little bit overlapping and some of the patterns of this type are this side and some of the patterns of this type and this type. So, therefore, still then also you make the margin of separation, but you also adopt some kind of penalty. So, that you may using this penalty you make this margin as wide as possible while taking care of this overlapping patterns.



The second problem can arise when this patterns are not linearly separable properly. So, in that case you can use something called a kernel function. So, you can use kernel functions. So, that you using the kernel if you represent this data is in terms of kernel functions they are quite linearly separable. So, therefore, you can adopt the original method that you used in the kernel space and do it.

In fact, you really do not have to do it there are certain kernel tricks using this kernel tricks you can you do not have to even go to the kernel space and do it using this kernel trick you can have certain method to get this optimally separating hyper plane. So, these are some of the references I have used. In fact, most of the examples that I have used are either from this one, this one or this one. In fact, I have followed the first reference as my guideline and prepared the lecture.

So, these are some of the conclusions. Classification maps the input to a qualitative output represented in terms of nominal and ordinal scale KNN decision tree, Bayesian classifier and rule based classifier are used only for classification purpose. Whereas, ANN we saw that it was used for regression now we saw that it can also be used for classification. In fact, decision tree also unless otherwise I mean the some you can have something called a regression tree as well which anyway we did not cover. So, then SVM we also studied about a new category of classifier which has to remember some of the training point training patterns it is called support vector machines. So, with this we give a brief introduction to the classifiers in the supervised training setting with this we wind up this lecture. Thank you.