Course Name - Recommender Systems
Professor Name - Prof. Mamata Jenamani
Department Name - Industrial and Systems Engineering
Institute Name - Indian Institute of Technology Kharagpur
Week - 06
Lecture - 28

Lecture 28: kNN Classifier for Recommender System

Hello everyone. Now we are going to continue our discussion on Content Based Recommender System models, where we are specifically going to talk about another very powerful yet simple classifier which is called KNN classifier. So, with this content now let us go ahead and try understanding what this K nearest neighbor classifiers are. In fact, while talking about our introduction to machine learning, we have already talked about all this in somewhat little detail, little briefly. Now we are going to talk about them in detail. So, now coming to this KNN classifier, it is based on the learning by analogy.

By comparing a given tuple with the training tuple that are similar to it. For example, when a new user when a new user when a user is coming to your site your website or your e-commerce site and you have some item, then you have to compare what this user has already seen in the past and compare with all the items and finding the similarity value you can recommend. So, which means these similarity values will be computed then and there at the time of giving the recommendation. So, here again the training tuples are described by n attributes.

Each tuple represents a point in a n dimensional space. In an unknown tuple a K nearest neighbor classifier such as the pattern space for K training tuples that are closest to the unknown tuple. So, which means this new item will be compared to all the items that the user has seen in the past and find out whether the user is likely to like that item or not. And if out of K more number of tuples more number of tuples based on the similarity based on the higher similarity value are near to the target item then it will be recommended otherwise it is not. Now, this closeness is defined in the form of formally in the form of certain distance or similarity metric.

And about this distance and similarity metric we have already discussed at depth in our earlier lecture. And in those lecture we understood that coming to similarity values higher the similarity higher is the closeness higher the distance lower is the closeness. So, which means using both distance and similarity we can do the task, but we have to little careful how do we whether you are using a distance function or a similarity function. So, now let us just have a probabilistic view of what I have told just now. It attempts to estimate the conditional distribution of response variable y given the feature vector x.

Then that and then classify a given observation to the class with higher estimated probabilities. So, given a positive integer k and a test observation x 0, KNN classifier first identifies k points in the training data set that are closest to x. So, in our training data set in our training data set let us say there are n observations 1, 2 up to n observations feature 1, feature 2 up to feature p and there is a response variable. So, the new item has what feature 1 to feature p. Now this has to be compared with each of these items.

$$\Pr(Y = j | X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = j)$$

So, when you compare this what do you get? You get a similarity values. So, similarity 1, similarity 2 and so on similarity n. So, total n similarity values. So, out of those n similarity values you choose top k and identify k items over here. Now from this k items you look at the response variable of the k items let us say you have chosen 5 items and in that 5 items which are closest to the new item you have chosen 5 items and out of that 4 items are liked by the user and 1 item is not liked by the user.

Director	Lang	award	type	Likes?
Dir1	English	no	Drama	no
Dir1	English	no	Scifi	no
Dir2	English	no	Drama	yes
Dir3	Hindi	no	Drama	yes
Dir3	Other	yes	Drama	yes
Dir3	Other	yes	Scifi	no
Dir2	Other	yes	Scifi	yes
Dir1	Hindi	no	Drama	no
Dir1	Other	yes	Drama	yes
Dir3	Hindi	yes	Drama	yes
Dir1	Hindi	yes	Scifi	yes
Dir2	Hindi	no	Scifi	yes
Dir2	English	yes	Drama	yes
Dir3	Hindi	no	Scifi	no

Then what is your what do you say? You say it belongs to the yes category that which means it will be liked by the user or in other words you have to if you have total k number of neighbors you count wherever the value of the response variable is some j. So, in our example continuing with our example from the last class the response variable was yes and no. So, 5 yes 2 no and total number of let us say observations we took is total number of neighbors we took is 7. So, this is the probability out of this is the highest. So, you predict.

Now continue with our example suppose there is a new movie with the details director 1 in the yes and drama. Now we have to predict the class variable whether you would like the movie or not. So, how do we go about it? We have to find out the distance of this vector with all this taking this 4. Now how do we find out the distance or similarity? There are many measures, but the problem here is if these values are not numeric. So, the first task is to represent in numeric form.

Director	Lang	award	type	Likes?
Dir1	English	no	Drama	no
Dir1	English	no	Scifi	no
Dir2	English	no	Drama	yes
Dir3	Hindi	no	Drama	yes
Dir3	Other	yes	Drama	yes
Dir3	Other	yes	Scifi	no
Dir2	Other	yes	Scifi	yes
Dir1	Hindi	no	Drama	no
Dir1	Other	yes	Drama	yes
Dir3	Hindi	yes	Drama	yes
Dir1	Hindi	yes	Scifi	yes
Dir2	Hindi	no	Scifi	yes
Dir2	English	yes	Drama	yes
Dir3	Hindi	no	Scifi	no

Director	Lang	award	type	Likes?
100	100	01	10	01
100	100	01	01	01
010	100	01	10	10
001	010	01	10	10
001	001	10	10	10
001	001	10	01	01
010	001	10	01	10
100	010	01	10	01
100	001	10	10	10
001	010	10	10	10
100	010	10	01	10
010	010	01	01	10
010	100	10	10	10
001	010	01	01	01

The target vector (Dir1, Hindi, Yes, Drama) = (100, 010, 10, 10)

English	1	0	0
Hindi	0	1	0
Other	0	0	1

Dir 1	1	0	0
Dir 2	0	1	0
Dir 3	0	0	1

Drama	1	0
Scifi	0	1

Yes	1	0
No	0	1

Now look at this what kind of now while finding those you have to be really careful about the nature of the variable that we are considering the measurement scale which you are using. Now what are the measurement scales in each? All these are qualitative in nature and they are nominal. All these are nominal not even ordinal all these are nominal. Now if it is so, how do I use some distance or similarity measure which basically require the numeric data? So, one of the way in which we have already covered that we can go for binary formation which also popularly known as one-hot encoding. So, we start encoding for English, Hindi and others.

Where are they English, Hindi and others? We have in this language we have three things English, Hindi and others. So, English can be represented 1 0 0 Hindi can be 0 1 0 and others 0 0 1. Similarly, director 1 directed through you obtain drama, sci-fi we have only two types movies drama and sci-fi. So, 1 0 0 1 yes is this. So now, we represent each record in terms of this encoding.

Director	Lang	award	type	Likes?
100	100	01	10	01
100	100	01	01	01
010	100	01	10	10
001	010	01	10	10
001	001	10	10	10
001	001	10	01	01
010	001	10	01	10
100	010	01	10	01
100	001	10	10	10
001	010	10	10	10
100	010	10	01	10
010	010	01	01	10
010	100	10	10	10
001	010	01	01	01

$$\text{Cosine}(\overline{X}, \overline{Y}) = \frac{\sum_{i=1}^{d} x_i y_i}{\sqrt{\sum_{i=1}^{d} x_i^2}\sqrt{\sum_{i=1}^{d} y_i^2}}$$

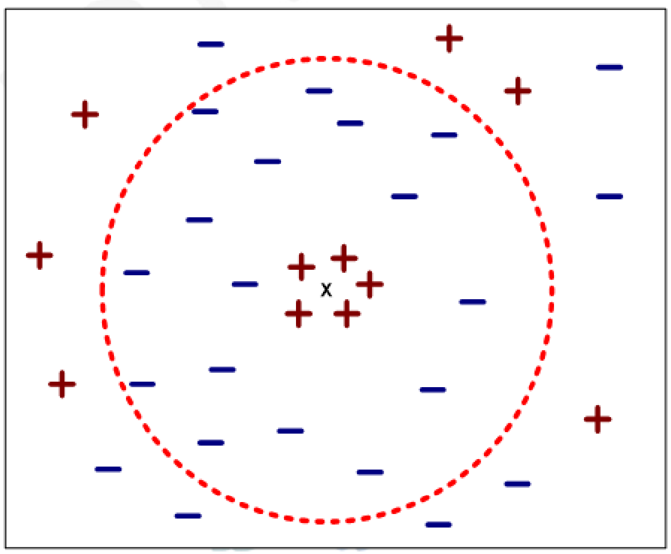
Director 1, director 1 language is English, English is this language is 1 0 0, award no. So, award is award was yes no type. So, no was 0 1. So, this is 0 1 and type type was drama and for drama we had 1 0 1 0. We made a encoding of this qualitative value which was basically in nominal scale.

So, now after we do this we also encode the target vector. So, now if you would like to compare target vector with that of the each of the record how do we do it? Let us say director the first record was 1 0 0, 1 0 0, 0 1, 1 0 and this one this one this is the first record and this is the target record. Target record was 1 0 0, 0 1 0, 1 0, 1 0. Now, this is the first record, this is from the training and this is the target. How do we find out the distance? How do we find out the distance or similarity? Finding the distance also there are methods, but we are if we use cosine similarity what do we do? This is your x, this is your y.

So, element wise element so this is basically your normalized dot product. So, element wise element you multiply. So, how many places you have 1? These places are matching both are 1 and these places it is matching both are 1. So, basically you will have 2 here and below how many 1s are here? 1 2 3 4. So, here also we have 1 2 3 4.

So, this you will be getting a similarity value. So, similarly with respect to each you will be computing the similarity, you will find some values and out of those similarity you will be choosing the top k. And within that set of top k now you will be counting how many yes and how many no that is likes. So, if the number of likes more number of yes is there then you say this is item will be recommended otherwise it is not. So, these are some of the issues of k nearest neighbor.

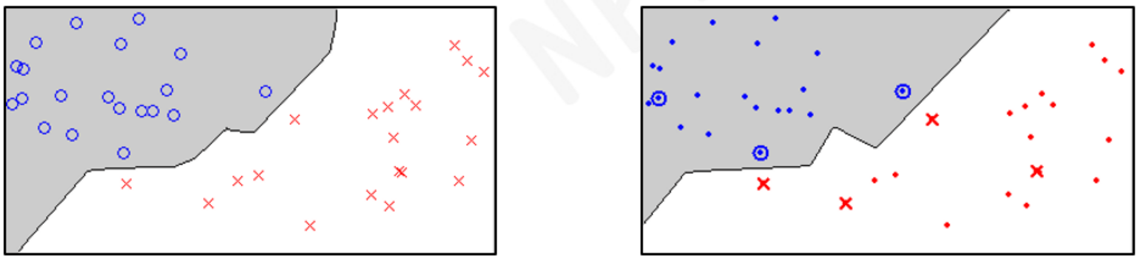
Finding the value of k the number of nearest neighbor to retrieve. Then choice of the distance and similarity metric this basically depends on the measurement scale normalization procedure that you have used and so on. So, all these preprocessing part we have already discussed. Then next issue is how to deal with computational complexity like when the size of the training data set is very large or the dimension of the data is very large number of rows and number of columns. Now coming to the finding the value of k typically you have to decide how large it should be.



Now if we make it very small there can be issues. If you make it very small probably you will have I mean the you may include some noise points. And if you have very large like this, this is supposed to be a positive example. And there are in this small neighborhood if you say this appears to be positive. Now if we go for this large neighborhood number of negative examples are more.

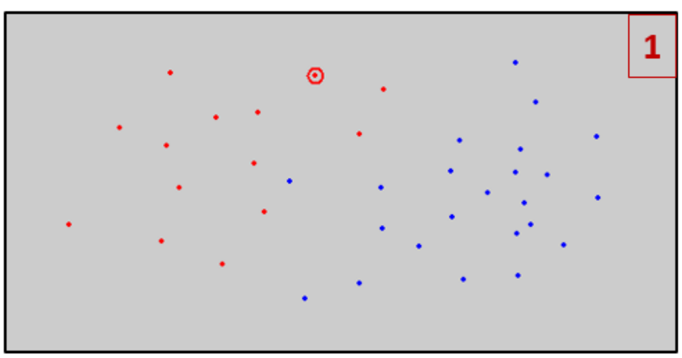
So, how do we decide? This decision is again has to be taken online. Why because online only you are determining? You have to take it offline because online itself you cannot determine the value of k. So therefore, this is the rule of thumb which is used, but there is no proof that you have to. So, in the offline setting when you are conducting experiments find out the value of k for which you are getting the good result.

So, you may choose that. This may be a good starting point in this regard. What is n number of records? KNN is a lazy learner. We already know what a lazy learner is. In case of eager learners like tree, tree based methods, Bayesian methods, rule based methods that we have seen in the last few lectures and ANN, SVM all this they are trained and what do you keep in the name of learner is the model parameters. So, which means and when you do the task in the online setting you simply have to use those parameters in the model that you have created and get the output.



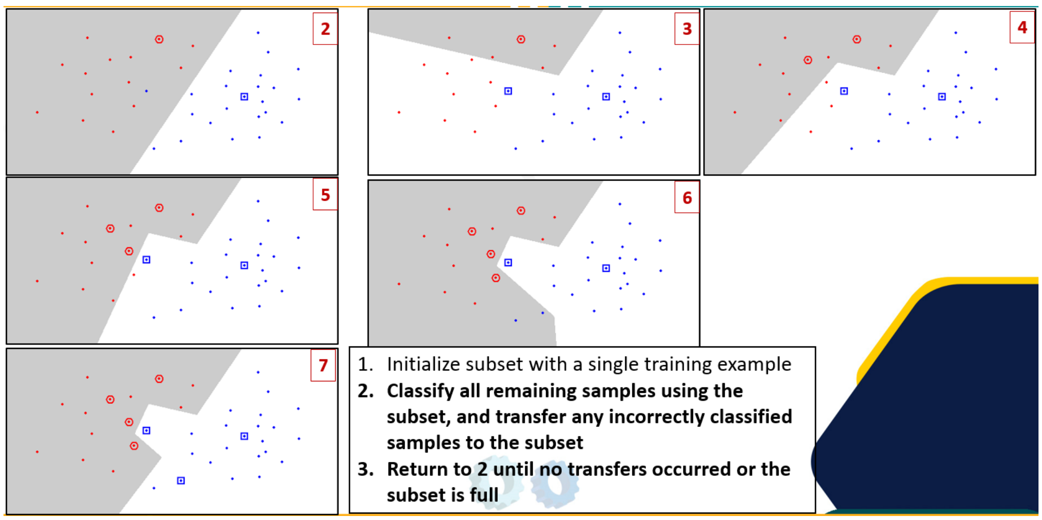
Original data Condensed data

But in case of lazy learner no such things happen. So, they simply store the training patterns and only at the time of prediction you can use all those. This is going to be problematic in our case because we may have large number of such patterns. So, online if you are trying to compare them and recommending based on you know in one e-commerce site probably people will not wait how long you are going to check, how long you are going to take. So, you have to be actually simply keep these training patterns and stay, do nothing, not training as such.



So, at the time of classification only you have to use some distance measure and compare. So, there are certain approaches for reducing this high computational complexity. So, one such approach is condensing. So, using this condensing approach you reduce the number of training example and keep some representative of those. So, that whatever decision you could have taken taking all the training patterns the same decision you can take with this representative sample.

So, in this it turns out that in this condensed data we have instead of so many points we have only 3, 4, 5, 6, 7 points which can be used. So, this is from a large data set now we have concentrated with only 7 points. Now let us look at that is condensed nearest neighbor algorithm. In case of condensed nearest neighbor algorithm you start with you have you maintain two sets. Let us say X is the all patterns and you some other set Z you start with nothing.



So, in the first iteration itself you put one of this random sample out of this one random you put it here. So, this one is added. This one is added. Then you do what? Then you classify all the remaining sample using the subset and transfer any incorrectly classified sample to this subset. So, what do you mean by this? Look at this.

Given this point which was there in Z, Z has this one point. Now with respect to all the remaining points how many remaining points? So many all these blue dots including this all those remaining points you find out the distance with respect to this. And find out the one where the distance is low yet it is quite close to it. So, in that sense we now discover this. Now your Z contains one red pattern and one blue pattern blue pattern this pattern.

What happens next? Keep these two inside Z. So, rather we will name these patterns. So, let this be A, let this be B. So, we have now A and B in our set Z. Now with respect to A and B both these have certain class.

So, A belongs to class let us say C 1, this belongs to class C 2. Now with this with the remaining you try comparing with A and B. These are training patterns you know that what is the class variable associated with them. Now when you find the distance for example, here is one point call it C. So, this point when you compare to B and A this supposed to be closer to A supposed to be closer to A this is closer to A.

So, which means it got incorrectly classified with respect to this new set of data. So, you got it incorrectly classified and because it is incorrectly classified you now bring in C here. So, now with this if you see adding A we partition it into two parts this entire data depending on which one is close to which other.

So, this was the partition. Now we have 3. Now with respect to this one when we take the distances of rest of the points we can again partition it further. So, when we partition it turns out that there is another point another point which is close enough to this one this one our A this was B this was C. Now we have discovered another point. So, which is A B and C and we have discovered another point D which is supposed to be close to A, but it turns out that it has become close to C.

So, now you transfer D to this set as well. So, now A B was there A B was there now C is there and what is the class of C? Class of C was C 2 same as that of B and now you transfer D what is the class of D? Class of D is same as that of A. So, by adding D with respect to C now you have made one more partition this was your original partition now you have made one more. So, this way you keep on comparing with this Z and find out with the remaining points and find out whether they belong to red class or they belong to blue class. So, continuing this process now it turns out that after this after this adding D this was A B C and D. After adding D it turns out that the points which were supposed to be A and D red they have become close to this one this point and this point.

So, they got added then sorry after this we have to come to this is fourth iteration this is fifth iteration. So, this point got added. So, let call it another point. So, after that you added this one more point and so on. So, continuing in this manner now in our final set we have this A B C D let us call it E F and G.

Now, we have more E F and let us say G. Now, for D what is the class? Class is again C 1 for E what is the class it is again C 1 for F it what is the class it is again C 1. For G what is the class? What is the class? It is again C 2. So, now when the new pattern comes instead of comparing with everything all the patterns all these dots that is available you only choose this is my condensed set. I use this for example for comparing. So, in online setting also my now I do not have much problem because my set is comparison for the comparison purpose my subset is very small.

So, these are my references and I have taken the example from this PPT as well and a lot of things from Han and Kamber, Charvagarwal etcetera and something from this one as well. So, these are the conclusions KNN attempts to estimate the conditional distribution of response variable Y given a feature vector X and then classify a given observation to the class with highest estimated probability. So, it is a lazy learner it has few issues like this what is the value of K what is the distance and similarity matrix and how to manage the computational complexity. For choice of K we saw there is certain rule of thumb and for finding this distance and similarity we saw we have to first bring it to certain numeric form in case they are discrete. Then we have to choose the subset of this entire feature set using certain algorithm called condensation algorithm.

There is other methods as well, but through which you find out a subset of sample that can be used instead of the entire data set at the time of classification. So, this lazy learner problem is relieved up to some extent. Thank you.