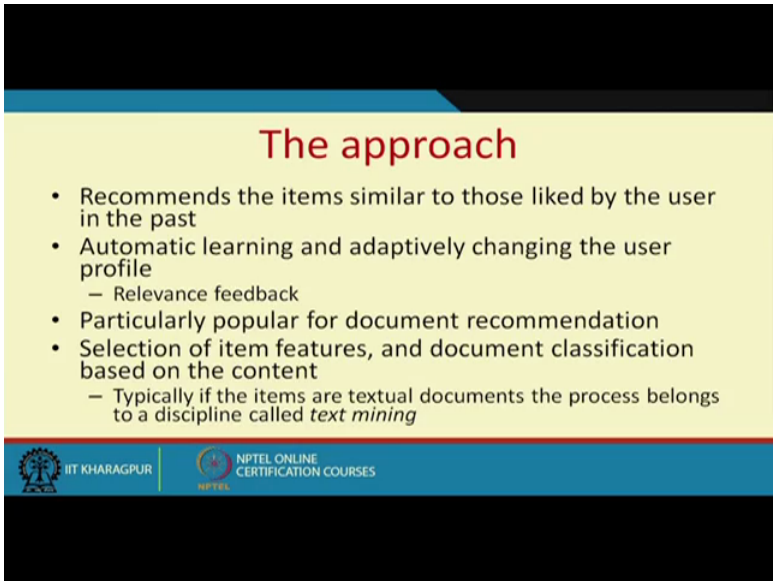
E Business
Professor Mamata Jenamani
Department of Industrial and Systems Engineering
Indian Institute of Technology Kharagpur
Lecture-53
Content Based Recommender System

Welcome back, hope last lecture you have understood what are various types of recommender system, in next few lectures we are going to discuss about specific recommender system. We really cannot cover recommender system itself is a big topic and we really cannot cover everything. So as has been made clear from the beginning of this series of lectures, we are simply talking about various decision supports applications and we are going to see how modelling techniques can be used for decision support. Here what is the decision support situation?

Instead of a actual sales person who could have assisted a user on a website, we are developing a system who can assist the user, who can provide the decision which could have otherwise taken by the salesperson for showing certain products or managing the managing the activities of Pacific users by suggesting you additional things which he may not be aware of in online environment through this recommendation generation process. Specifically in the today's class we are going to look at the Content based recommender system.

(Refer Slide Time: 2:16)

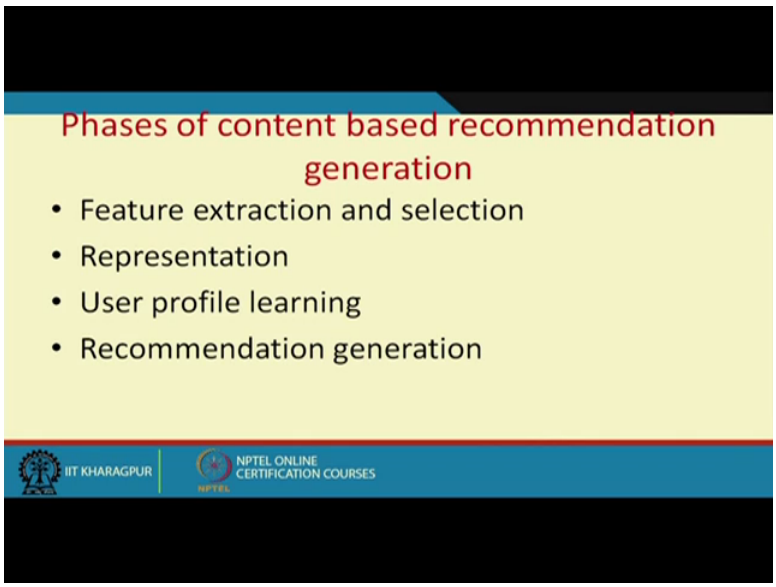


So in this approach, the items which are similar to each other and some of these items are already seen by the users, they are recommended. So what data it will require that what items, the items the user's past history of search on various items, it has this data. And these items

how they are similar to each other, based on their content is another data source okay. Now this kind of systems today I saw about recommender system book, tomorrow I may be seeing something else, today I was interested in sports because some match is going on test match some 20-20 is going on and I was interested in sports. And it is not that my interest is only limited to sports, I might be seeing something else, I might be interested in movies, I might be interested in the (())(3:44) I might be interested in what is happening in the finance world.

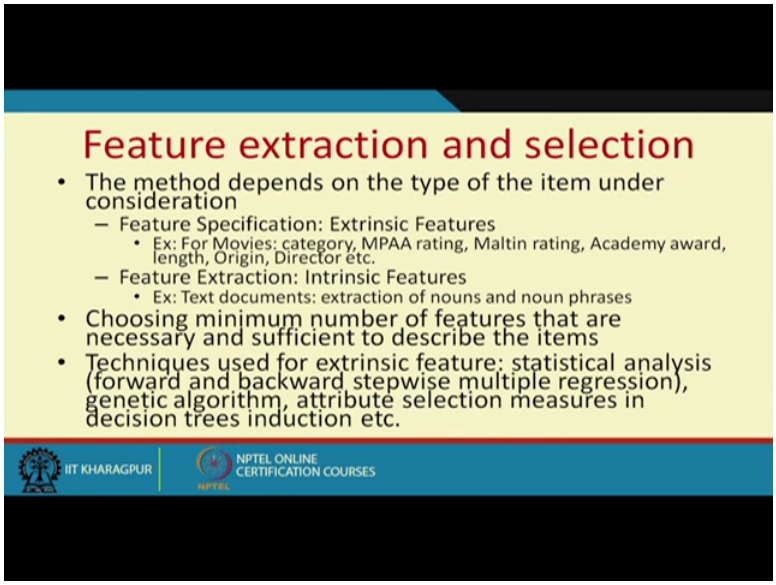
So okay, so therefore looking at my past preferences and how it changes, this content based recommender system should not only see my reference and suggest the item, but it should also understand how exactly my interest is changing over the time and suggest the item accordingly okay. So they are these kinds of systems are particularly popular for document recommendation like that of your news items, et cetera. It tries to find out tries to find out the similarities between these text documents by understanding its content in terms of the keywords it uses and so on, so extracting the features which are related to the text within the item so some text mining ideas can be incorporated to find out the features.

(Refer Slide Time: 5:18)



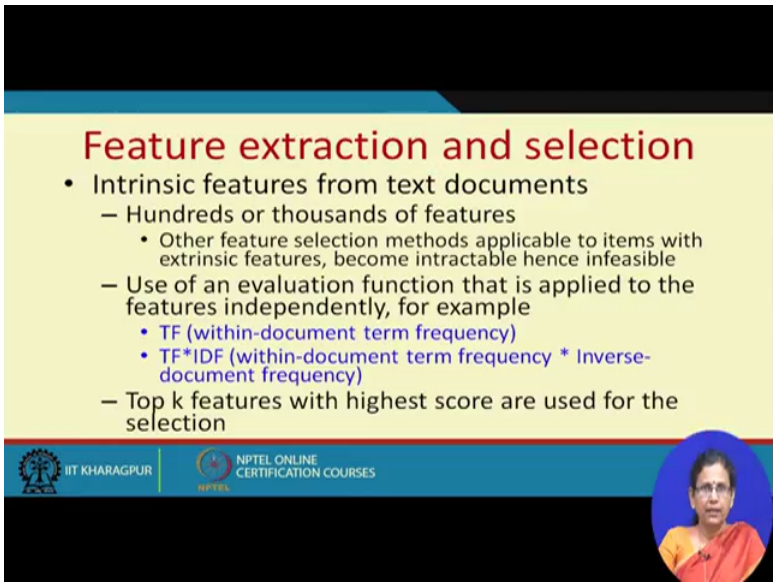
So these are the phases involved in content based recommendation generation, feature extraction and selection whose feature? Feature of the item, feature extraction and selection, representing the feature in the right manner, user profile learning that is how the preference of the user can be captured and represented and some using some model to learn the learn the user profile then generate recommendation based on the item details and user profile.

(Refer Slide Time: 6:00)



So first task is your feature extraction and selection, so these methods depends on the type of item under consideration, these features can be of 2 types, they can be either extrinsic features which are quite obvious and already data is available for that for example, in case of a movie the movie category is MPA rating, it has various other ratings, whether it is Academy award winner or not, its length, its origin, its category, et cetera can be used as features. Similarly, in case of intrinsic the features can be intrinsic which means it can be derived, it is not obviously available somewhere. As I was telling you, in case of text documents extracting maybe the nouns and noun phrases can constitute your features. So here the concept of the various ratings can be derived from the text document itself.

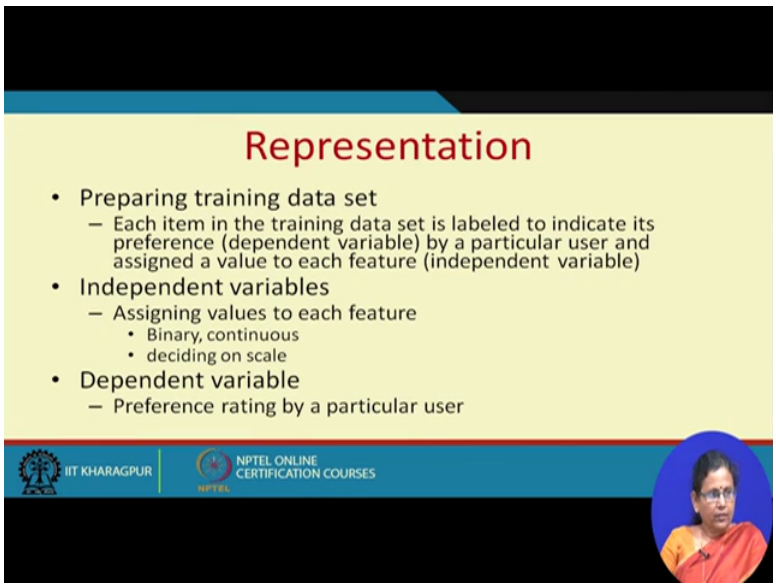
(Refer Slide Time: 8:13)



Though we are not discussing, this can be your term frequency inverse document frequency, et cetera can be used. Now choosing the minimum number of features that are necessary to describe the item are also important to figure out. So finding the right kind of features and right number of features is again an iterative procedure and some machine learning tools and statistical tools can be used for this purpose. As I was telling you, the extrinsic features where the extensive features are obvious and available readily, intrinsic features specifically for text documents have to be derived, there are many problems associated with it.

Typical there will be hundreds of hundreds and thousands of features, now which of those because if we take each word as features there will be thousands. So which of these actually we have to take is a problem so that is what I was telling, so use of maybe you can use some kind of evaluation function like TF rating that within document term frequency and TF into IDF rating that within document term frequency into inverse document frequency. It is not limited to these 2, this can be many more, there can be many more but these are very frequently used features. Now Top k features with higher scores can be used for describing the item, then representing the item,

(Refer Slide Time: 9:14)



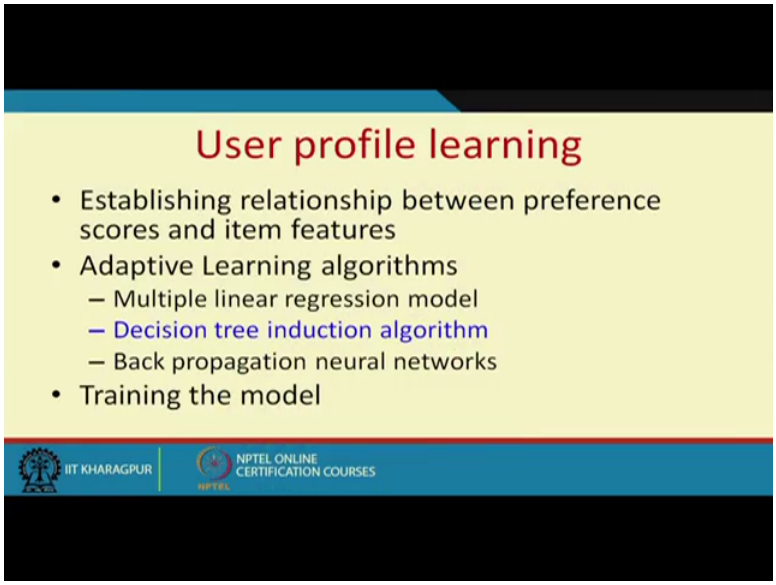
So the idea is to prepare here as I told you, you have to learn the user preference model. So to build that model you need some kind of training data, so the model will be first trained with this training data then it will be tested with, I mean whatever is the past available data based on that part of the data will be used to train the model and to validate whether the your model is properly trained or not you will be using some sort of test data okay, so first task is to prepare the training data set. The training dataset basically will have 2 parts, the first part will

be some sort of independent variables, which will be typical the products features seen by a user, I mean see what are we going to build and what are we trying to train, we are trying to train the user preference model.

So which means we have in our hand the past data of the user regarding his browsing pattern of various products. If for example if the products are movies then all the past movies that user has seen will be my data input, so if I have let us say 100 movies out of which the user has seen 50 movies then what will be my dataset? My dataset will be the movie features will be my independent variable, my dependent variable will be whether the person has seen the movie are not. So out of my list of 100 movies, 50 movies I will be having the details of the movies as input as my independent variable and whether he has seen or not as a binary variable as my dependent variable.

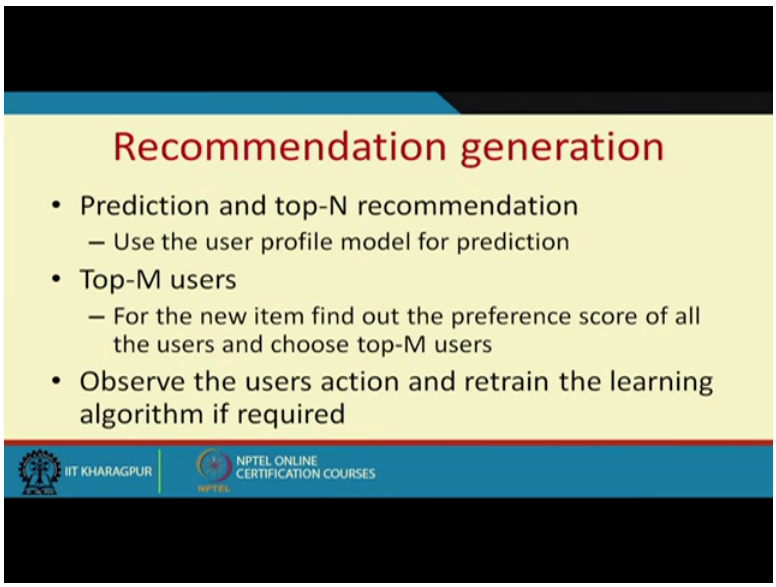
So I have to identify the dependent and independent variable for a specific user then this independent variable has to be assigned the values of the feature vector you have derived then the dependent variable will be the preference rating of a particular user, this preference rating can be again explicitly given by the user or it can be implicitly derived for example for example, just now we were discussing whether the person has seen or not is something which can be used as an implicit feature to decide the users rating. Or if the user has specifically given rating after seeing the movie then that rating can also be used as your dependent variable. Irrespective of that the point that is made clear that is I am trying to make clear is you have to have a dataset where the input the independent or input variables are the item features and the output variable is user preference.

(Refer Slide Time: 13:14)



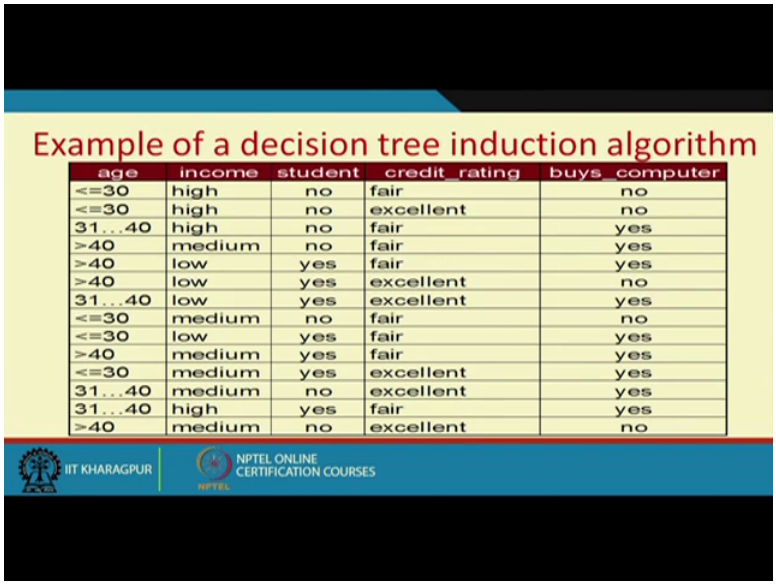
Then next task is user profile learning, so it is about establishing a relationship between the preference score and item features. Many algorithms can be used for this, one of these algorithms the decision tree induction algorithm with the items with some specific known features explicit features can be used to understand this particular model.

(Refer Slide Time: 13:42)



And these are some of the things like we have already discussed, this can be used for prediction and Top-N recommendation. Top-M users can also be done, for a new item you can find out the preference score of all users and choose the Top-M users; this can also be one of the approaches. But anyway, we are not simply going to look at the recommendations on one kind of situation prediction kind of situation.

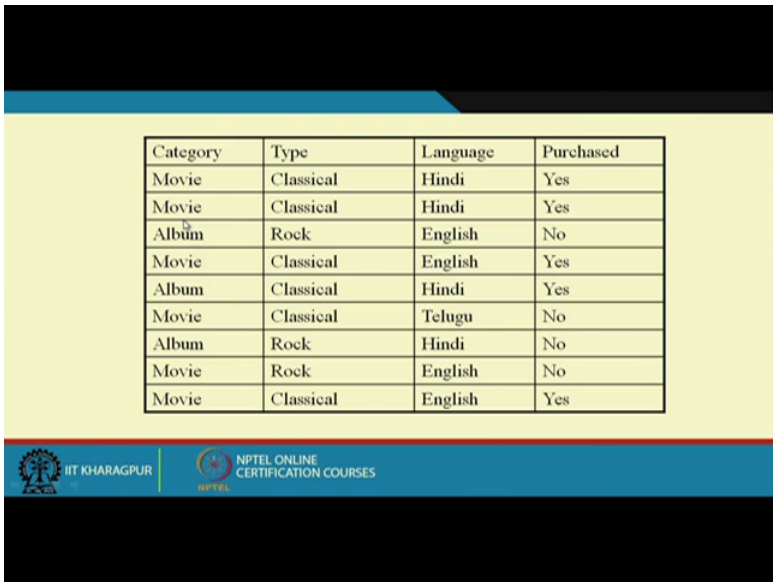
(Refer Slide Time: 14:27)



age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

Let us see situation where buyer is coming to your store online store and you are in the process of selling a computer to the buyer and the buyer's characteristics for various buyers is shown here and based on his characteristics, his age, income, whether he is a student, credit rating, whether he has good credit rating or not, based on that you have to decide whether he will be buying a computer or not and accordingly you will be giving him the preference.

(Refer Slide Time: 15:21)

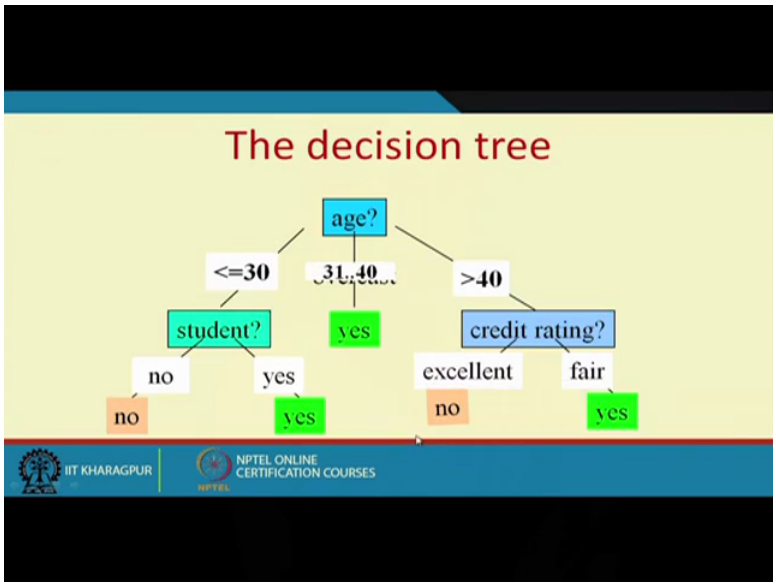


Category	Type	Language	Purchased
Movie	Classical	Hindi	Yes
Movie	Classical	Hindi	Yes
Album	Rock	English	No
Movie	Classical	English	Yes
Album	Classical	Hindi	Yes
Movie	Classical	Telugu	No
Album	Rock	Hindi	No
Movie	Rock	English	No
Movie	Classical	English	Yes

This example can be extended to a movie situation, where for a specific buyer let us say here there is another situation in which the same model can be used in this situation as well. Here you have various items, their types, and their languages and users purchase history with the item, same model can be used for providing preference of a single user as well. So now we

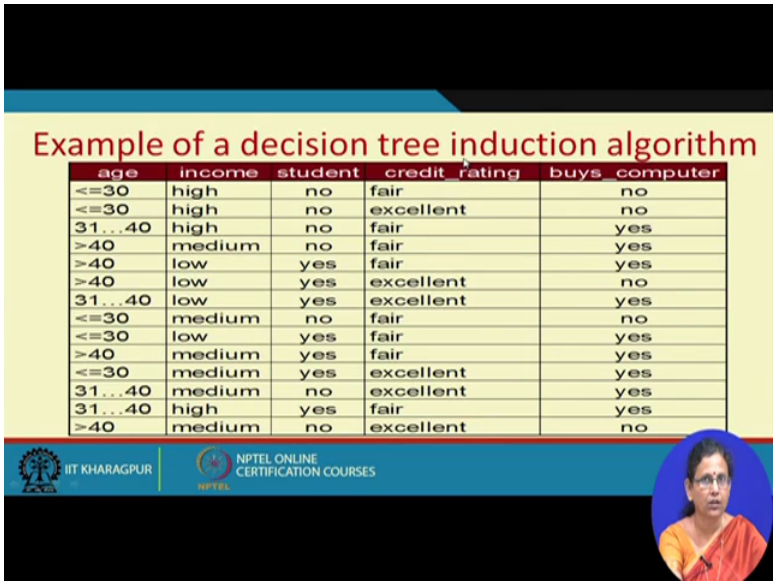
are trying to figure out if a new person comes in, based on his features whether you are going to predict an item for him or not. This is one example of induction tree algorithm and this is truly not the item feature, this is just the purpose of showing how induction tree induction algorithm works, we are showing this but later on we will give you one example problem which you should solve yourself to find out how products features for a specific user can be used to provide the provide this decision whether he will be buying something or not.

(Refer Slide Time: 16:46)



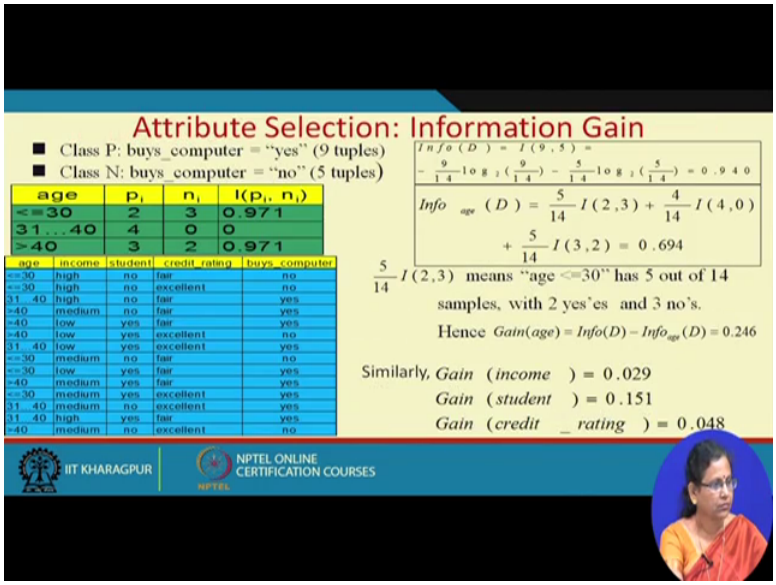
So based on this data you can construct something called a decision tree, which might be of this form. Like what is at the root there will be age, if somebody comes who is a... First of all you will be checking what is his age, if the age is between 31 to 40 then based on your past data you know that he is definitely going to buy it, so you will suggest the item. If his age is less than equal to 30 then you will find out whether he is a student or not, if he is a student then you have to then you will be sure that he will be buying the item he will buy the computer. If his age is greater than 40, you will look at his credit score, is his credit score is excellent he is not going to buy the item, and if his credit score is fair, he is going to buy the item.

(Refer Slide Time: 18:18)



Look here, the nodes represent these variables; age, income, whether he is a student or not, these are the input variables. And the leaves of this tree tells you his purchasing decision, so each of this part will tell you whether there it ends up in buying or ends up in not buying or ends up in buying. Now the next question is how do I construct such a tree? So this decision tree induction algorithm can be used to construct such a tree.

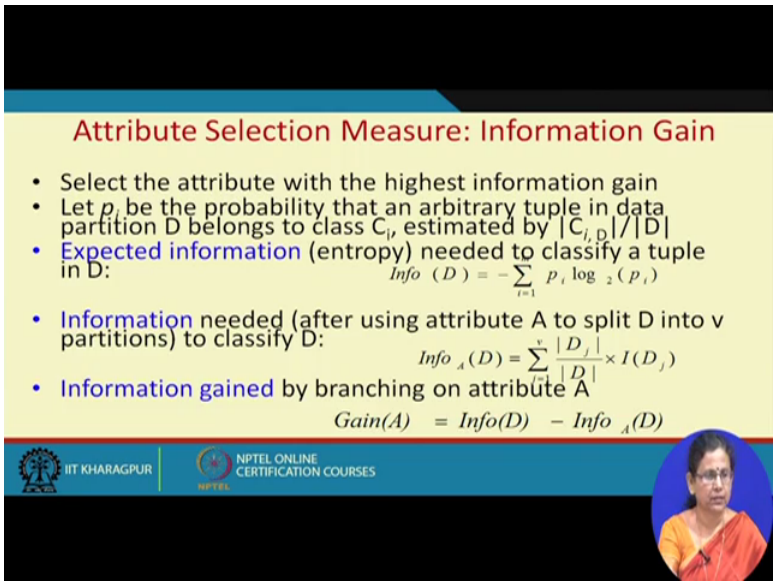
(Refer Slide Time: 18:59)



So this in fact all these things we can come back little later, in fact it is about partitioning the data, by reading this it is really does not going to make much difference, we are going to understand the example, we have to partition the data. What we have done here, we have partitioned the data first based on this age variable into 3 parts; this is one part of the data,

this is second part of the data, this is third part of the data, what was our data? Our data was this. So based on this variable, everybody who is less than or equal to 30 makes one part of the data, everybody who is in between 31 to 40 they make another class and greater than 40 another group, so data is partitioned into 3 parts.

(Refer Slide Time: 20:14)

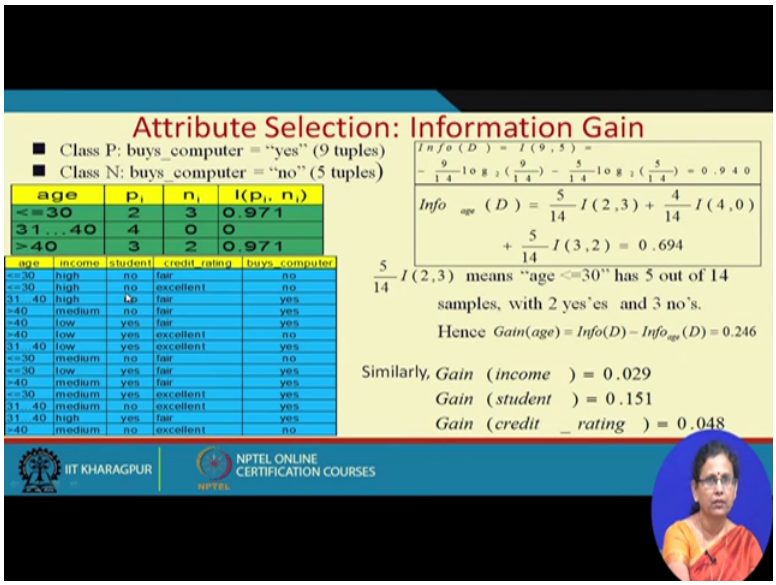


Now the question is why age? We had so many different 3 different variables, so why exactly we are choosing age and not income, student or credit rating? To see that why we are choosing age we have to learn about attribute selection measure, so there are many attribute selection measures in fact, people who have by chance taken a course on data mining, they must be knowing that there are many measures of partitioning this data but out of that we will now be choosing only one just to show how exactly this kind of decision support situation can be implemented.

So the measure for partitioning this data set we use as your information gain, this information gain is calculated in this manner. First of all you have to find out what is the expected entropy needed to classify a tuple in the entire dataset D, then you find out the information content of D. This is a standard formula for finding out the information content or the expected information or entropy, so using this formula you will be finding out the information content in the entire dataset, then what do you do? You for each of the attribute each of the input attribute, here the tributes were age, whether he was a student or not, his credit rating and so on. For each of the attribute you find out, for that specific attribute what exactly is the information needed.

So the difference between this information content of the original dataset minus the information content in breaking the dataset into individual components based on a specific attribute, difference between these 2 gives the information gain, whenever the information gain is the highest you choose that.

(Refer Slide Time: 22:32)



age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31...40	4	0	0
>40	3	2	0.971

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

So we continue with that example, in this example this is the data that we have already seen. So the information content of the entire dataset where 9 of the tubles were with Yes decision and 5 of the couples with No decision, here you can find out count which all the couples, each of the row is a tuple, so each of the tuple how many tuples were with yes decision? 3, 4, 5, 6, 7, 8, 9, so 9 tuples were with yes decision and 5 tuples were with No decision. So using the expected information entropy formula we find out this is the information content in the entire dataset.

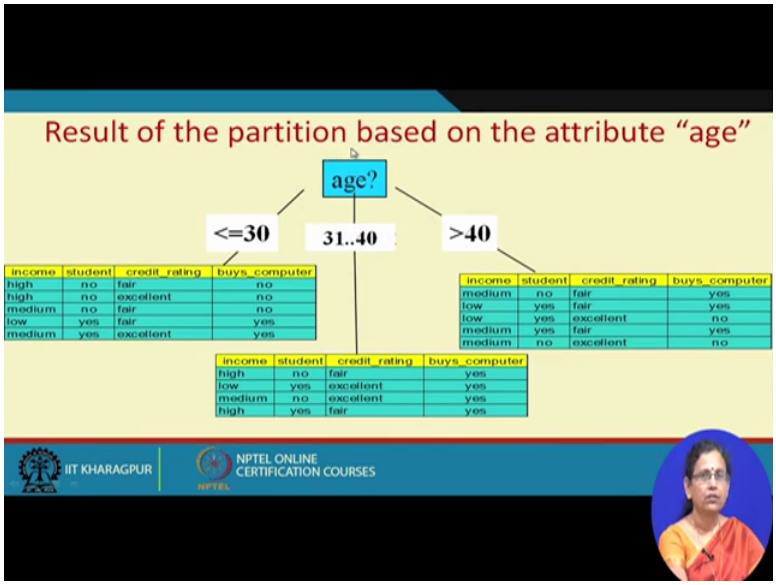
Now, if we partition it with the variable age using this formula, you will find out information content in case of information content in each case of each of the partitions. So what were the partitions? Partitions were whether the age was less than equal to 3, age was between 31 to 40 and age was greater than 40. So there were 3 partitions; for each partition for the first partition let us say less than or equal to 3, less than or equal to 30, 1, 2, 3, 4, 5 total 5 number of 5 number of entries are there and out of these 5 number of entries that are 2 No and this is here is one 3 No and 2 Yes okay 3 No and 2 Yes okay, 2 Yes and 3 No.

So this information gain for first partition, this is the information gain for the second partition, this is the information gain for the third partition, now how many tuples belongs to

this? It is total 5 out of total 14 tuples it is total 5, so relative importance of this is 5 by 14, relative importance of this is 4 by 14, relative importance of this is 5 by 14. So taking this expectation of this of this partition, taking the expected value for this information content of this partition over all these 3 parts of the dataset turns out to be this much, this computation you do by yourself this turns out to be this much. So information gain is difference between this point 940 minus this information content due to age, so difference between this and this.

So it is not only that, with respect to each of the input variable; income, student and credit rating, you have to find out these values. Out of these 3 values, whatsoever is the highest, in this case it is age, you know take that as the root. So the key question that we are trying to discuss if we have to take provide decision support for whether he will be buying a computer or not whether to show the advertisement or to show the product related to computer, first task is based on our data we had construct we had to construct this tree. Now if we have to construct this tree, the question was out of these input parameters, out of these independent variables which one should be at the root?

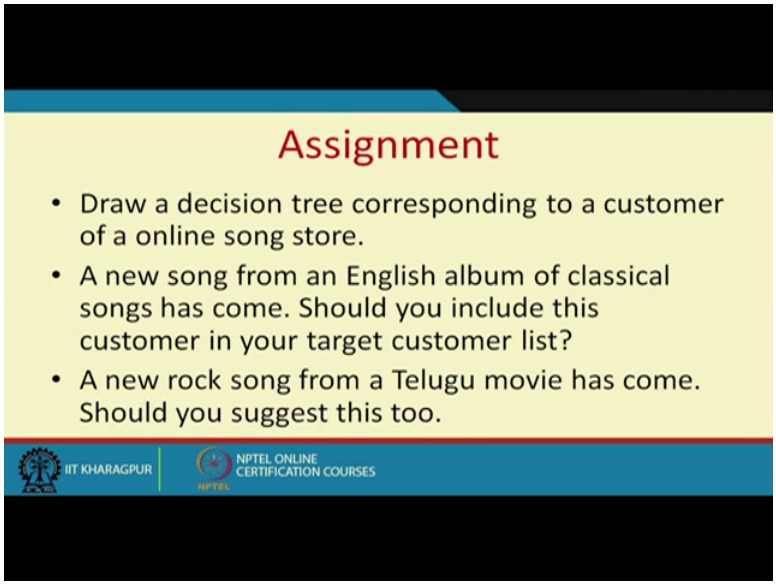
(Refer Slide Time: 27:27)



Now here we found out the procedure by which you have decided which one will be your root. Now once you know the root, you partition this is what I was telling, this was 3 partitions. And if we continue this process, you end up for this partition you get students as your next node and give partition. Here you do not have any other node; you directly reach at the decision, here you find out the credit rating as the next and continue. For example here, here it can be you can further make 2 partitions so you have to find out, out of these 3 which is the right variable and it turned out to be students. And here you do not have to make any

further partition because all these output values are same, so once this is satisfied you get this, by this process you can develop a tree which is a minimum height as well okay.

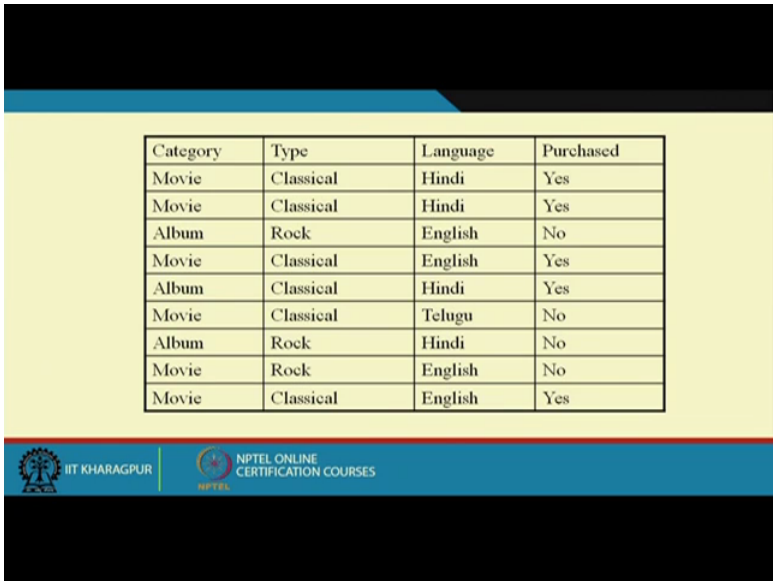
(Refer Slide Time: 28:31)



Now this idea of decision tree can be extended for content-based recommender system. Now see, this is the assignment for you is like this, the questions will be asked based on this assignment. You draw a decision tree corresponding to a customer of a online song store, a new song from an English album of classical forms has come, should you include this customer in your target customer list? What kind of decision-making scenario it is? It is Top-M user, by Top-M user we mean if a new item has come who are those users who can be targeted with, who can be made known about this product through advertising or by some other method.

Now, each of these users have their own product browsing history, based on that what you do? You build a profile model for each of the users okay. Now this model can be used for predicting the weather the particular user will be buying this or not. So out of those all the users available to you from your customer base, you can find out who are those customers who can be targeted with okay.

(Refer Slide Time: 30:15)



So look at, this is the data for a single user, now you say whether a new English album of classical song has come, whether this particular user should be targeted with or not? Now a new rock song from a Telugu movie has come, whether you should suggest this one to this customer? For these 2, you have to now find out whether you should be recommending those 2 to your user or not okay. So prepare with this, thank you very much and we move to a new topic next class.