Business Analytics & Data Mining Modeling Using R

Dr. Gaurav Dixit
Department of Management Studies
Indian Institute of Technology, Roorkee

Lecture - 36
Classification and Regression Trees- Part I

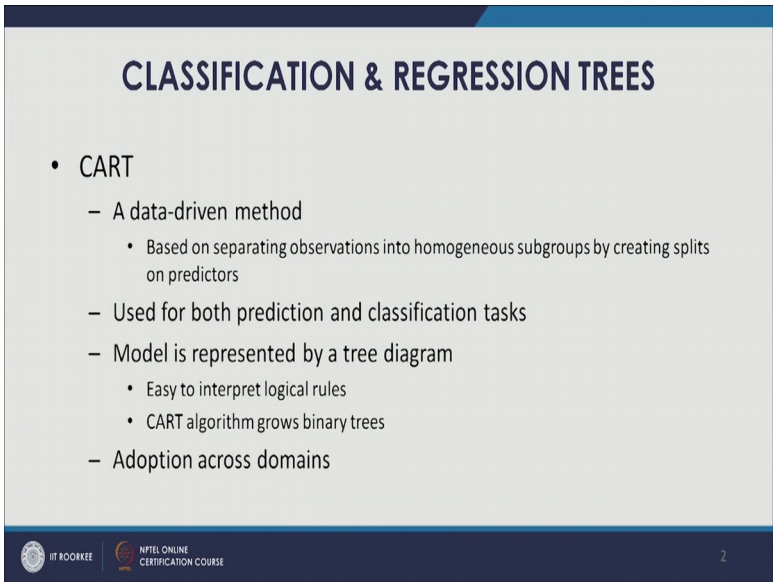
Welcome to the course business analytics and data mining modeling using R. So, in this particular lecture, we are going to start our discussion on classification and regression trees. So, let us say start. So, in short; this is also known as cart. So, many of these statistical software packages and data mining packages, they have they generally implement this particular procedure this.

So, this is the more common procedure for classification and regression tree and the more general name for this kind of technique is decision trees and under this we have this particular algorithm cart. So, this is a flexible data driven method. So, where in we do not actually have any conditions or like assumed functional form between the outcome variable and set of predictors. So, those kind of conditions that we generally have in statistical techniques that is not applicable here.

So, you would also see the data driven method and based on. So, how it works. So, it is based on separating observations into homogeneous subgroups by creating splits on predictors. So, as we had done one exercise in the starting lecture where we had the observation we had the observation for the sedan cars owner and their income and then and we had the data on their household area and you and the outcome variable was ownership or owner or non owner. So, there one of the method as an example; we had tried out we were creating rectangles in that particular graph and that was actually a similar approach what is actually done in cart.

So, the number of observations that we see; so, they are the partitions are created right.

(Refer Slide Time: 02:25)



So, we separate those observation into homogeneous subgroups. So, the same thing that we had done in the very fast lectures very first lecture series the introduction lectures that we had. So, here the classification and regression trees we generally separate observations into homogeneous subgroups and by creating splits on predictor. So, depending on the values of the different predictor variables we find the best one to create 2 separate create partitions to create homogeneous subgroups and that is how we keep on keep on separating observation and creating homogeneous groups. So, this can be this particular technique can be used for both prediction and classification tasks as we will see in this lecture series.

Now, how the model is represented. So, typically the model is represented by a tree diagram so that you might be familiar with. So, generally we start with the root node and the root node; there we have to find out the predictor the best predictor which can create the optimal splits and based on a finding a particular predictor and a particular value a value for that predictor to create optimal splits based on that the root node start and this process continues for both you know left sub tree and right sub tree. So, model is represented by a tree tree diagram and easy to interpret logical rules.

So, out of this model once our tree diagram has been built constructed then the rules that we get out of this technique are very simple for example, if there are 2 variables like age and income. So, and we have to classify responder and non responder. So, out of this tree

we might get a rule like this if is greater than 25 and income is less than fifty thousand then class one. So, these are the kind of rules that might that we might get if age is greater than fifty and the income is greater than sixty thousand then class zero or class two

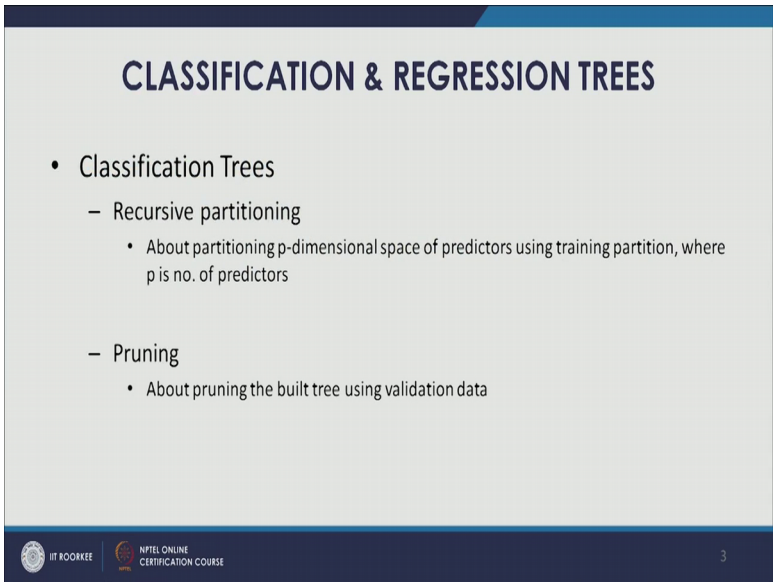
So, these are the kind of rules logical rules that we get and they are very easy to interpret so easy to identify observations easy to identify records which will meet these criteria which will meet these logical rules and easy to implement. So, classification and regression tree because of the logical rule simple logical rules that we get and because of the ease of implementation of these rules they have been applied in a variety of a wide range of domains and very popular across domain not just the analytics engineering medical and other domains also they have been used. So, and another important thing about this particular technique trees in general decision trees in general and classification regression trees cart in particular that very simple model.

So, therefore, when you first think about the problem when you first encounter a problem as we talked about the analytics challenge that you have to you know if they are if you are able to identify certain prediction tasks classification task then classification and regression tree is one particular technique where you do not have to think too much about the applicability.

And whether the assumptions the applicability the data various things that we have been trying out in previous lectures, you do not have to think too much you do not have to process or follow steps like other techniques and you can simply apply. So, this is going to be one automatic selection irrespective of the; a type of prediction tasks or classification tasks so different problem situations. So, this particular technique can be applied in different problem situations and also across different domains

So, let us discuss further the classification trees. So, we will start with the classification task first.

(Refer Slide Time: 06:50)



So, classification trees there are 2 typical steps that we have to follow to build a classifier model based on the cart procedure. So, typically the first step is recursive partitioning. So, what happens in recursive partitioning is as we have talked about that this is about partitioning. So, p dimensional space. So, if we have p number of predictors our set of predictors in our set of predictors we have p variables right. So, we are going to partition this p dimensional space of predictors using training partition using training data set and. So, this is how it starts. So, every time we will keep on partitioning and this process will go in any recursive fashion.

So, first partition and once the after we do the first partition will get 2 more partition if that is if we are if we are getting 2 partition sets then on those each of those 2 partitions will further apply will further apply this recursive partitioning approach and this process will keep on it will continue till we create you know all our partition that we finally, get till we achieve the pure homogeneous subgroups pure homogeneous partition out of this process.

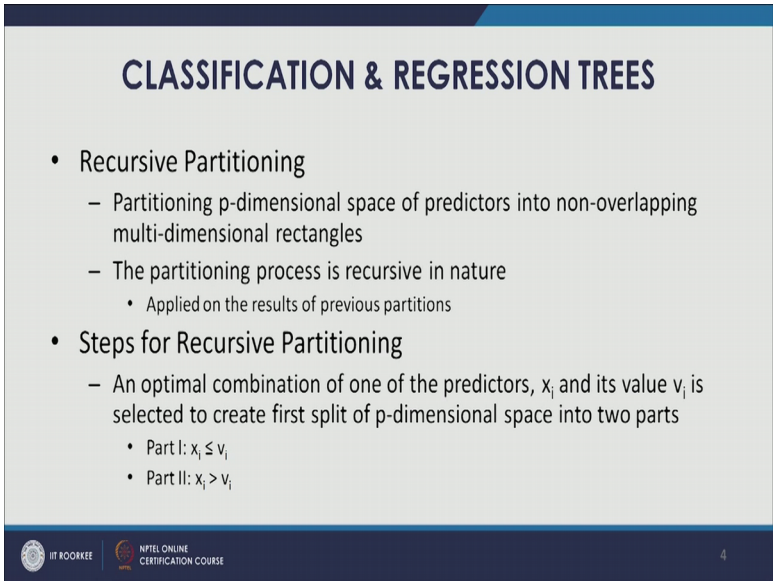
So, in this; so, first step recursive partitioning is mainly about partitioning the observation and mainly about creating the homogeneous subgroups in a recursive fashion the second important step in building classification trees is pruning. So, in pruning the build the tree that we have a build using training partition; so, because we keep on building that particular tree till we reach a pure homogeneous subgroups so that partially

fully full grown tree is going to classify all the observations correctly. So, that would actually over fit the data.

So, to have a good model which is not fitting to the noise and extracting and using the predictor information we will have to prune the tree back to a level where it is not fitting to the noise. So, pruning is that process. So, this is about pruning the build tree using validation data. So, the second partition that we deeply you know that we also discussed in initial lectures that the validation partition can also be used to fine tune or refine the model. So, in this particular technique classification and regression trees you would see that in the pruning step it is the validation data is that is used to prune the full grown tree or that has been built using the training partition; so, 2 main steps recursive partitioning and pruning.

So, in recursive partitioning, we will first partition the p dimensional space operators and do and do and will try to create homogeneous subgroups and once this process is done, then we like to prune the tree. So, that the model is the tree more diagram that we model that we have is fitting to the predators information and not to the noise.

(Refer Slide Time: 10:21)



So, let us discuss in more detail each of these steps. So, first start with the let us start with the recursive partitioning. So, in recursive partitioning number one what we do is partitioning p dimensional space operators as we talked about into non overlapping multi dimensional rectangles. So, of course, we are in the p dimensional space. So, with one

predictor it be partisan this particular p dimensional space, we are using one particular predictor the first step then the partition that we create this would be non overlapping and of course, they are even going to be multi dimensional.

So, we are going to create non overlapping multi dimensional rectangles up to first step. Now the as we discussed that partitioning process is recursive in nature; that means, the previous partition; so, the; we keep on applying this process recursive partitioning process on the results of previous partisans now for let us also understand the detailed steps for recursive partitioning. So, the first step is about and finding an optimal combination of one of the predictors let us say this predictor is x i.

So, and the value that is going to be used the split value that is going to be used to create a split to create partition is v i. So, we need to find out this optimal combination x i this particular particular predictor x i and a particular value of this predator v i. So, this has to be selected to create first a split of p dimensional space into 2 parts. So, a part one can be the all the values x i the values less than that vertical value v i.

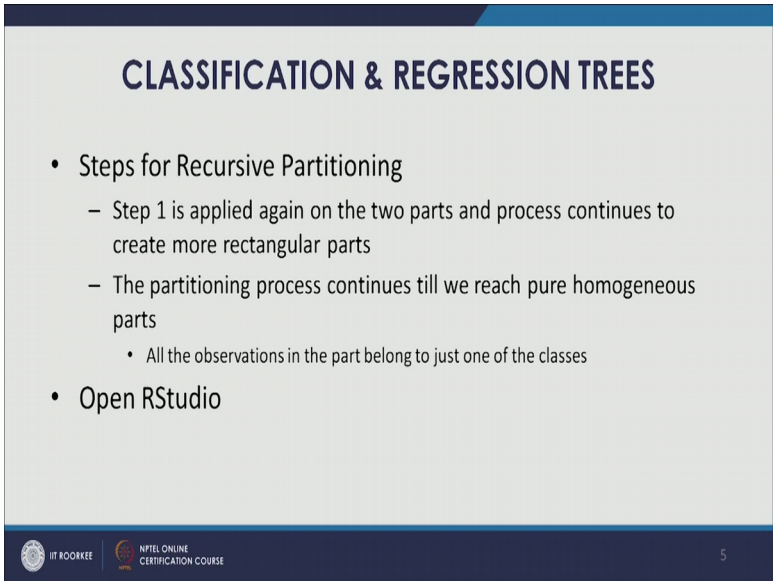
So, x i values being less than values less than or equal to v i. So, they would go into the part one and the part 2 the x i values which are greater than v i, they will go into the part two. So, using that particular value v i. So, all the observations all the observation or records which are having x i values less than or equal to v i, they will go into part one and all the observations are records which are having x i value greater than v i, they will go into part 2.

So, this is how we will create these 2 parts and also as we talked about. So, the selection of the particular predictor x i and the value the splitting value v i, it is has to be an optimal combination. So, next step in the next step the step one that we talked about that is applied again on the 2 parts.

So, when we talked about the recursive partitioning this is the recursive part the results of previous step again the same process is applied on those results. So, step one is applied again on the 2 parts and a process continues to more rectangular parts now once this process. So, this process will continue in the step three as you can see the partitioning process continues till we reach pure or homogeneous parts. So, till all the parts all the subgroups that we create all the partitions that we create they have the same

observation, they have observation belonging to the same class till that till we achieve that scenario this process continues you can see in the slide as well.

(Refer Slide Time: 14:09)



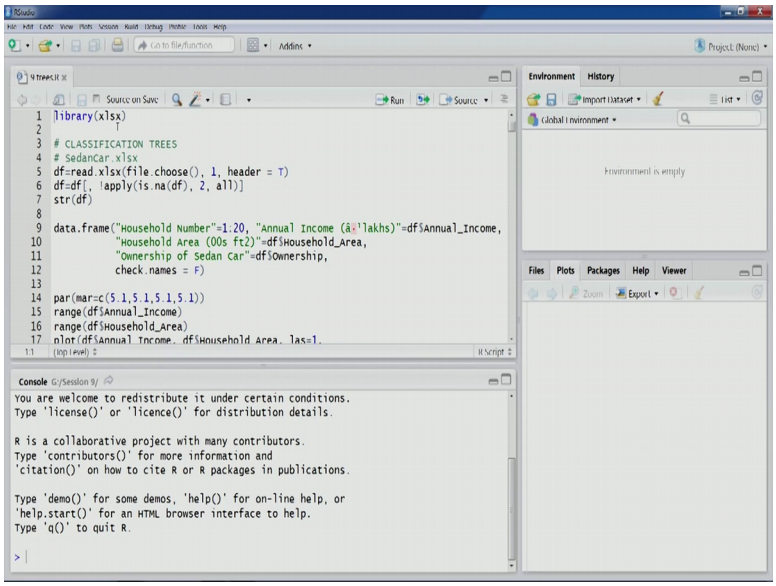
That all the observations in the part belong to just one of the classes.

So, the all the subgroups partitions rectangles that we create the observation that belong to those subgroups partitions rectangles they should belong to just one class one class. So, one of the classes; so, therefore, this process will continue till we reach that. So, this particular tree after we follow this these steps these three steps of recursive partitioning, but we will end up with a full grown tree which will be able to classify all the observations or the carts that are there in the training partition and it will have hundred percent accuracy and zero percent error because we have been able to we have been able to create groups homogeneous groups having members of just one of the classes.

So, let us understand this process regards the partitioning process through an exercise in R.

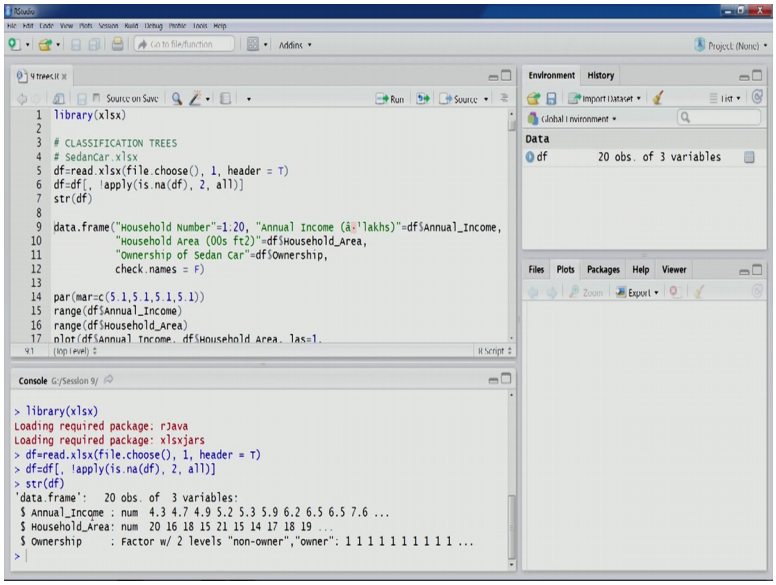
So, let us open R studio.

(Refer Slide Time: 15:17)



So, as we have data in excel formats let us load this particular library that we typically used to import the data set from excel files let us do this line.

(Refer Slide Time: 15:28)

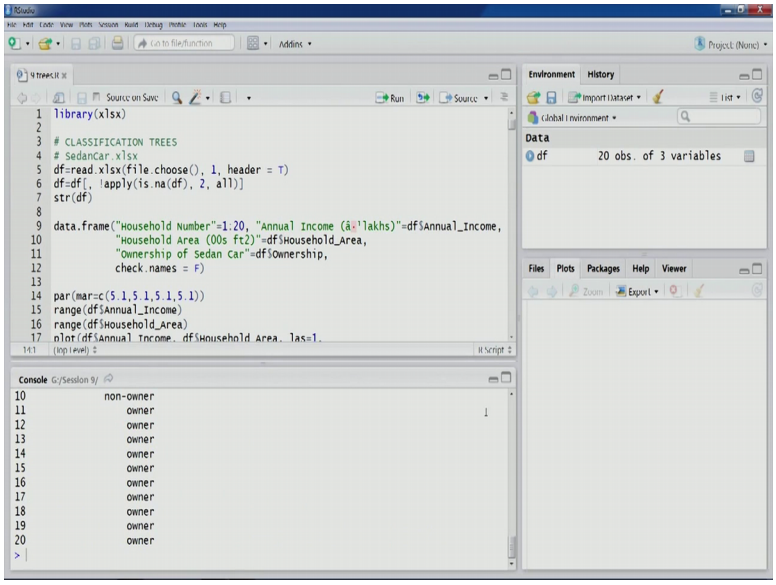


So, this is loaded. So, the particular excel file the particular data set that we are going to start with is sedan car data set that we have used in previous lectures as well. So, let us import this particular data set as you can see in the environment section data frame the data set has been imported into this data frame d f we have twenty observation of three variables. So, let us remove NA columns, if there are NA. So, once that is done let us

look at the structure of this particular data frame. So, as you can see this is the particular data set that we have used in previous lectures as well. So, there are three variables annual income household area and the ownership, alright.

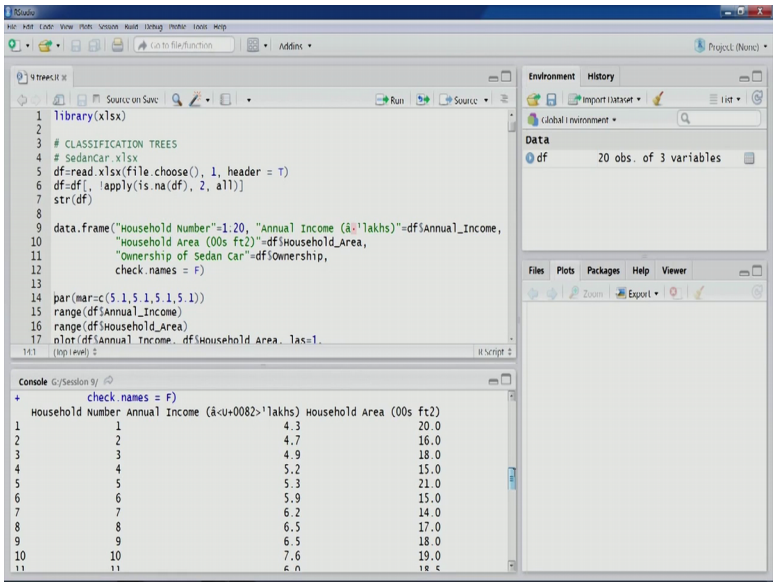
So, using these 2 predictors annual income and household area. So, we have 2 dimensional space here 2 dimensional space predictor space we have and we would like our outcome variable of interest is ownership. So, we would like to apply our recursive partitioning and that step of cart procedure on this particular data set. So, let us also look at the first few observations.

(Refer Slide Time: 16:53)



These are the first few 20 observations of this particular data set.

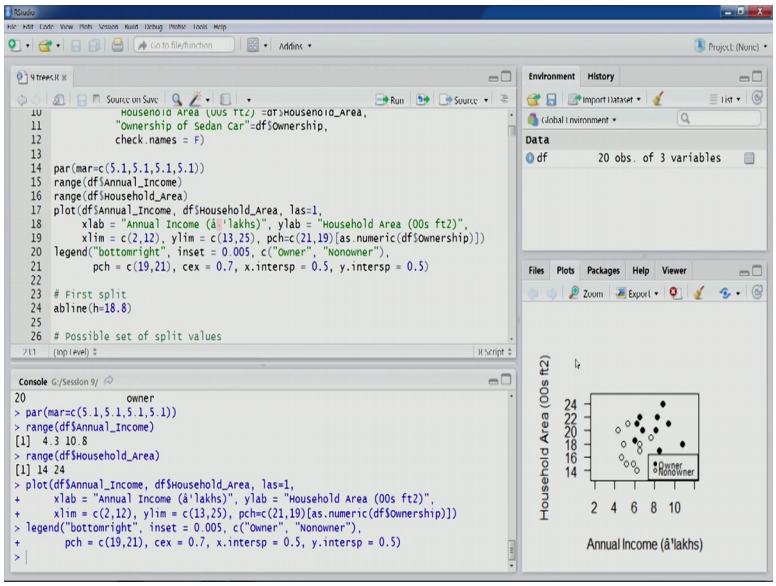
(Refer Slide Time: 17:07)



So, as you can see the income is in rupees, lakhs per annum this is annual income in rupees lakhs and the household area is in the square feet; hundreds of square feet. So, these are the unit of measurement for these 2 variables and then we would like to classify the different households unit of analysis is household. So, we would like to classify households into owner or non owner category; so, whether they own a sedan car or not.

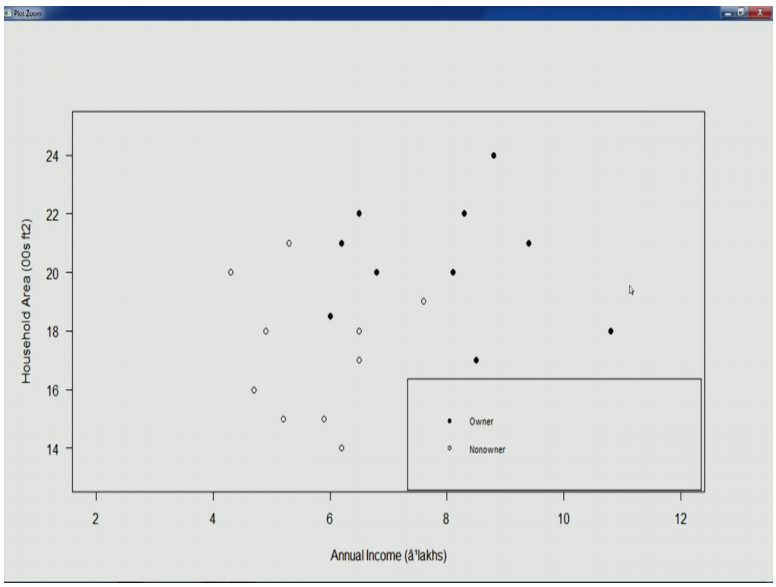
So, let us plot; let us create a scatter plot of this. So, let us first set some of the graphics parameter. So, as we have been doing using a power function and within that we would like to set this parameter margin; m a r; to these numbers to this. Now let us look at the range of these 2 variables annual income this is the range and the household area.

(Refer Slide Time: 18:04)



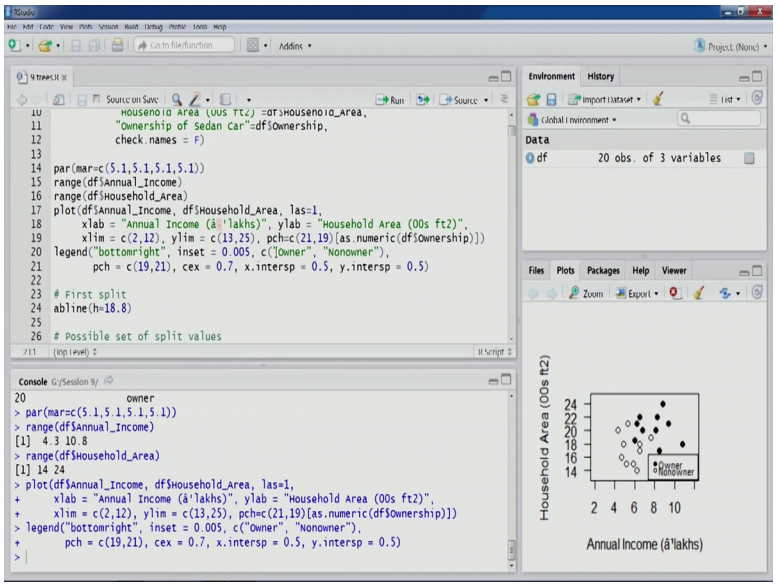
This is the range as we have been doing in previous lectures as well while we create plot we also specify a limit on x axis and a limit on y axis which is quite close to the values of these values or further as specified in the range of the variables to be plotted. So, you can see actually accelerate to twelve this is. So, the range of annual income is within this within this particular limit similarly for y axis y limit you can see 13; 25. So, the range for household area is within this limit fourteen twenty four. So, so that now let us create a scatter plot. So, this is the plot that we have it is also generated legend for this.

(Refer Slide Time: 18:53)



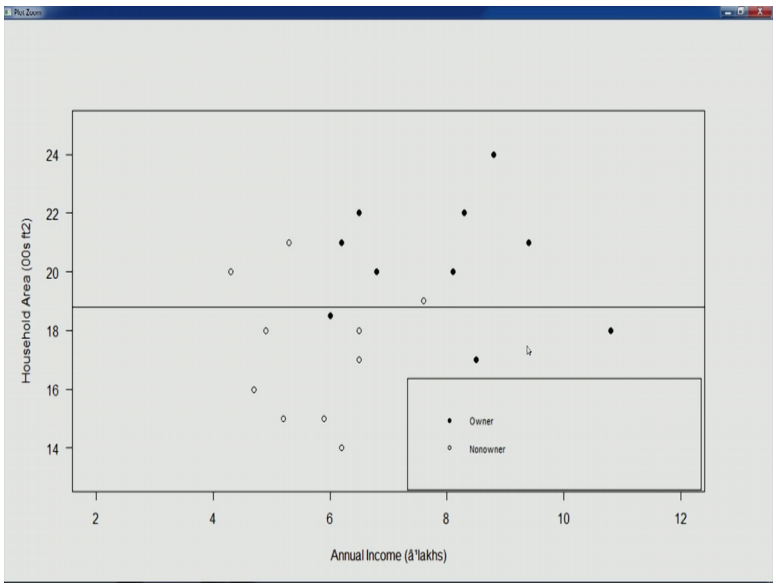
So, these are the twenty observations that we have and we would be applying the cart; cart procedure the card algorithm on this particular data set to create homogeneous subgroups or rectangles.

(Refer Slide Time: 19:05)



So, what happens in the first place? So, let us assume that first split is going to be on this particular value.

(Refer Slide Time: 19:24)



So, 18.8 that is this particular value; so, household area eighteen pointed this value is going to be somewhere around this somewhere close to 19 marks somewhere in between

18-20. So, this particular line is will probably go like this in between these 2 close points and we will get our first partition let us create first split. So, again these splits are hypothetical based on our visual inspection visual analysis that we see that probably this is split created using this particular value.

So, the variable that we have selected here is the household area and the value that we have selected is 18.8. So, out of 2 predictors that we had household area and annual income we have selected household area and particular value 18.8 as the optimal combination. So, of course, this is not from the result of applying the model cart procedure this is one example for illustration purpose we are going through.

So, if a household area and the particular where we value 18.8 that is the that is the optimal combination for operator value and this is how this is the kind of spirit that will have if we look at the upper rectangle here you can see here 7 observation belong to the owner class and remaining three observation belong to the non owner class, right. So, is not pure homogeneous, but most of the observations belong to the owner class, if we look at the lower rectangle the lower subgroup then we have 7 observations belonging to the non owner class and three observations belonging to the owner class. So, this is majority is with a non owner class, but again this is also not pure homogeneous.

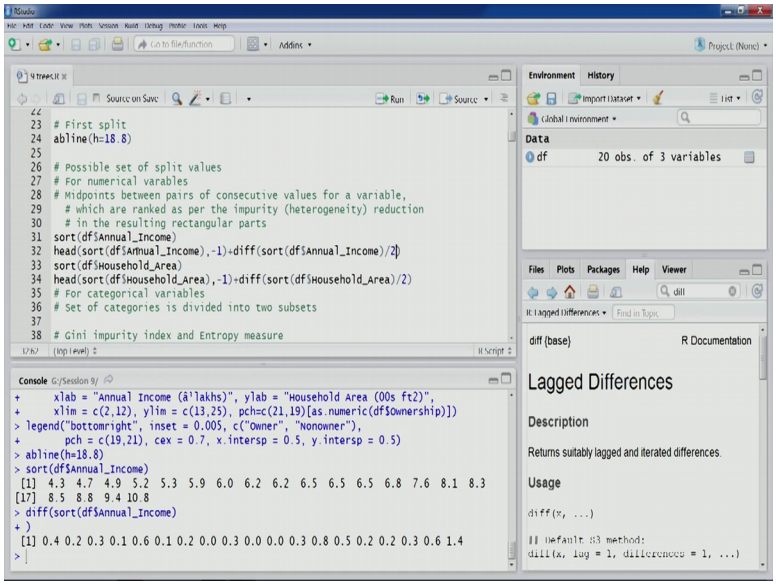
So, we will have to continue the process because as we discussed in recursive partitioning we will continue the process. So, we have these 2 partitions. So, on each of these 2 partition will apply the again we will do further partitioning till we reach to your homogeneous partition or subgroups. So, let us try understand; how we can proceed further. So, what could be another important another important concept here is to understand the possible set of split values right. So, what could be these possible set of values from which we can find out the optimal combination.

So, for example, if the variable is numerical; so, if the variable that is going to be selected the split variable if it is going to be if it is a numeric variable numerical variable then mid points between pairs of consecutive values for a variable they are going to be the candidates for possible split values. So, now, these midpoints can again be ranked as per the impurity reduction the heterogeneity and that could be there in the resulting rectangular partition.

So, that is how we can pick the optimal predictive value combination. So, for a particular predictor right the midpoints if it is a numerical variable the midpoints between pairs of consecutive values. So, they are going to be that those points are going to be the possible candidates and again we can further rank them with respect to their impurity in the resulting rectangular partition.

So, let us go through this process for the annual income variable. So, let us first sort this particular variable.

(Refer Slide Time: 23:13)



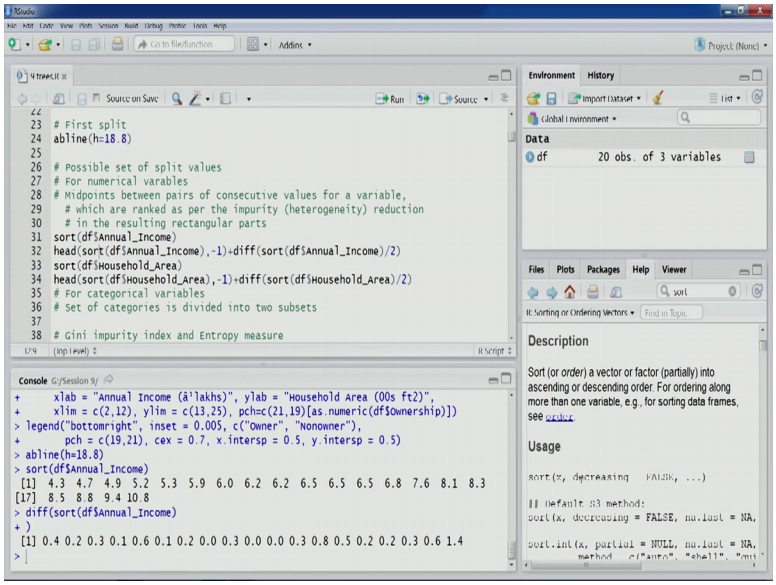
So, these are the value we have 20 observations. So, these are the value in sorted order you can see they are in increasing order. So, starting from 4.3 4.7 and then up to 9.4 and 10.8. So, once these values have been sorted we can start computing the midpoints. So, if we the once the value is sorted. So, we can the number of values are 20 here, right. So, in this case we will have we can create a diff here. So, this particular diff is going to give us on the sorted annual income value if we create this diff. So, these are the values that we get. So, in this diff these are the value.

So, annual income if we sort them and if we take it difference. So, if you are understand finding more about this particular function diff. So, what we are trying to do here is we are trying to compute the lagged differences here. So, it will; it is going to return suitably lagged and iterative differences. So, the default value as you can see here the lag is one. So, for annual income right the sorted values of annual income.

So, the lagged lag one values are going to be written and then we you would see it in the in the we also divide these values by 2 because for each value in the in the sorted sequence each value will add this the difference they will add the difference of you know half of the values half of the values for this particular output out of this output of this particular expression and in the first part you would see we have used sort let us also look at the sort function in the help function help section.

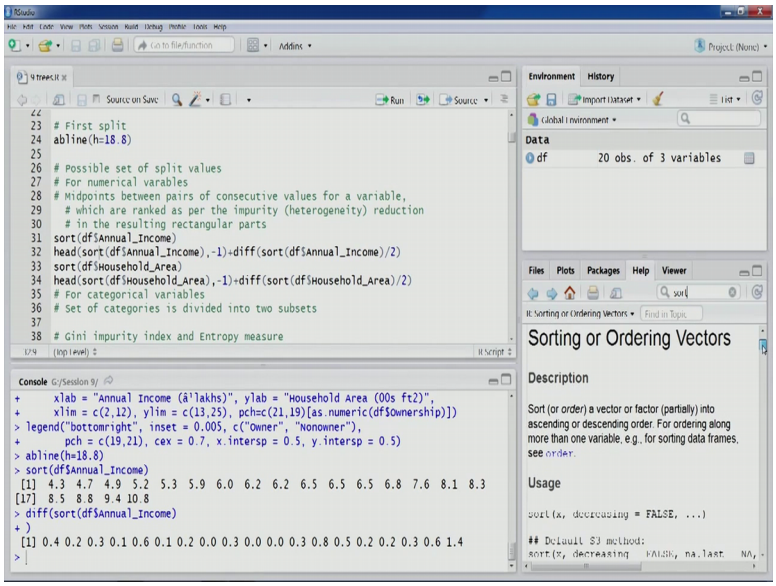
So, let us look at the arguments.

(Refer Slide Time: 25:32)

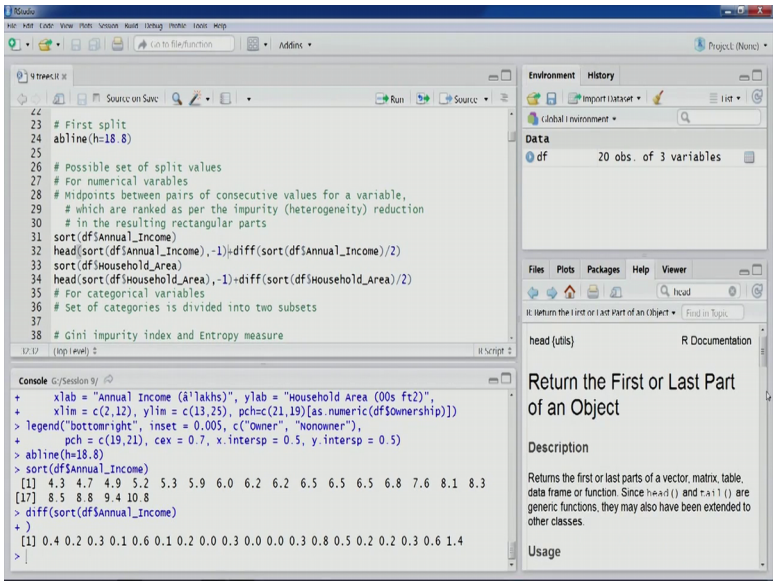


So, you would see sort the variable and after this we have we have taken head. So, let us look at these. So, the sort is going to create the order the typically that by default this is as you can see the raising is false. So, by default it is the increasing order.

(Refer Slide Time: 25:56)

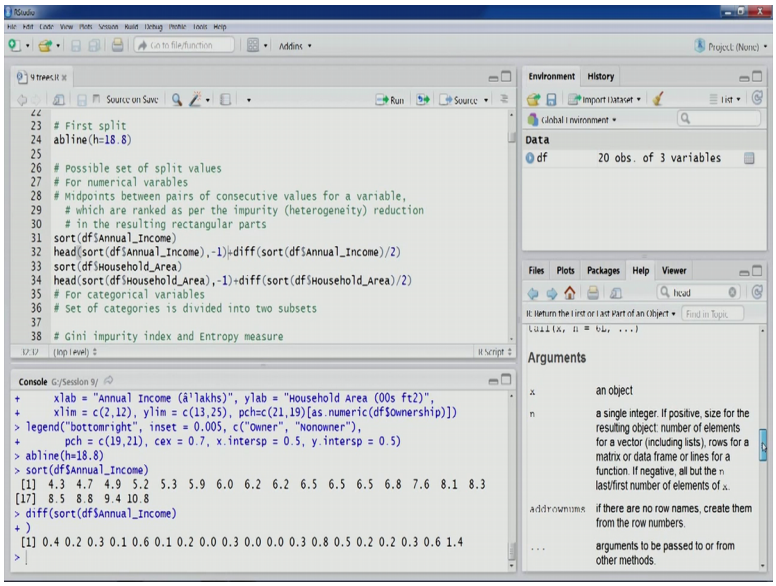


(Refer Slide Time: 25:58)



So, let us look at the head function the second argument we are interested in; you can see the second argument is n in this. So, this is a single integer. So, a positive size for the resulting object number of elements and you can see if negative all, but the n last or first number of elements of x.

(Refer Slide Time: 26:13)



So, once we take head of this particular and minus 1. So, the last value will not be included. So, you would see that you can see in this will have around almost 19 values you can say start from just like this output 4.3 to 10.8 and this starts from 4.3 to 9.4 right so on.

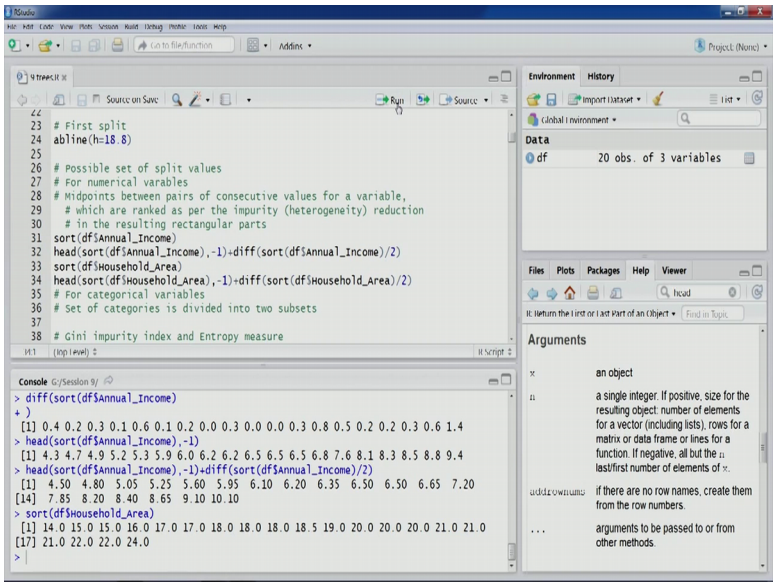
So, the mid points to compute the mid points; so, these first we compute this these values this particular series and then we add a you know the we take a lag and up using that lag we add this particular value here the half of that value we add here. So, let us compute this. So, this is what we get if you if you look at these values. So, they are the midpoints you can see 4.3 and 4.7; it is the midpoint values value for this particular; this particular here is 4.5 an x minus 4.7 and 4.9.

So, midpoint value is 4.8. So, from here you can actually see how we have been able to compute the midpoints. So, first we because we also wanted to rank them; so, first we have sorted; so, consecutive values. So, first we have sorted and then the last value, we had removed here using the head because the midpoints would be computed based on the based on the last 2 last value and then we have added the half of these numbers right half of these numbers sorted numbers and then by taking a lif diff.

So, diff will give will actually compute that delta that is there. So, if we look at this output of the diff here point four you can see the difference between .4 and 4.3 and 4.7 first pair this is 0.4. Similarly for others also the next one is 0.2 and 0.3. So, half of this

will actually you know give us the remaining part to compute the midpoint values for the same thing same process we can apply for all sold area as well. So, these are the sorted values in the increasing order.

(Refer Slide Time: 28:40)



So, once this is done as you can see the same code is there. So, again we leave out the last value and then we add that difference that is there, then half of that difference between each pair of you know consecutive values. So, we will get the midpoints; so, for these 2 variables the midpoints that we have this one for the annual income and then the household area also.

So, these particular set of points are the candidates possible candidates of split value split values. So, the in our process and because the partitioning process when we try and identify a particular combination optimal combination of predictor and the value of that predictor we will have to try out these many combinations for annual income we have around 14, 15, 16, 17, 18, 19; so, 19 values midpoint values that we have. So, this has to be 19 and then another for household area another 19 value. So, out of these 38 predictor value combination we will have to find out the optimal one and that is going to be used for our first split.

So, we will stop here and we will continue our discussion on this finding possible set of split values for categorical variables in the next lecture.

Thank you.