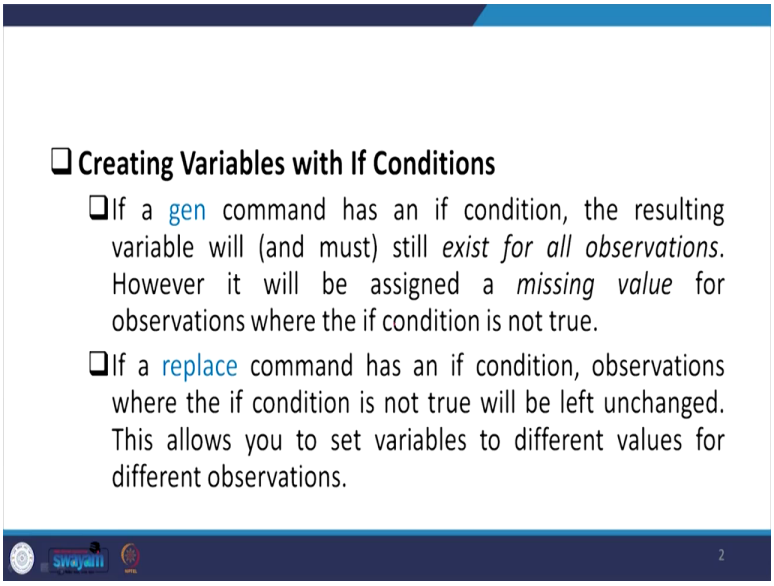
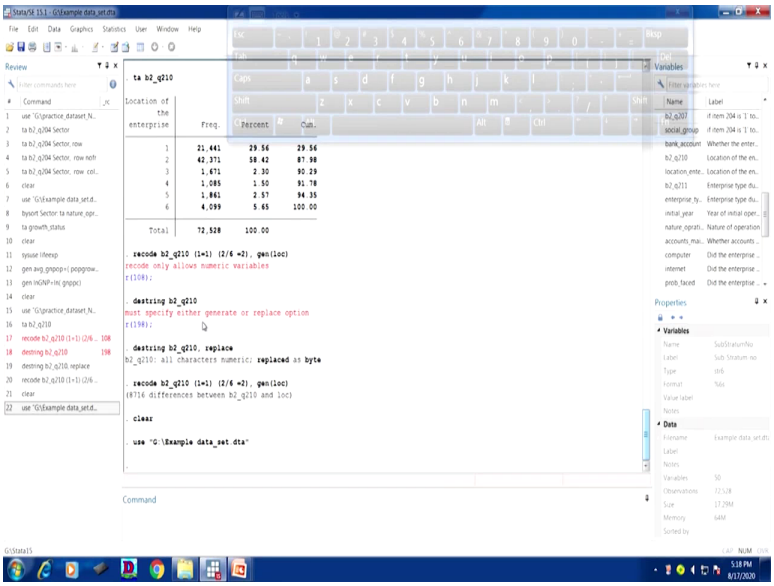
Handling Large-Scale Unit Level Data Using STATA
Professor. Pratap C. Mohanty
Department of Humanities and Social Sciences
Indian Institute of Technology, Roorkee
Lecture No. 19
Tabulation and Creation of New Variables in Stata – III

Once again friends welcome you all to the NPTEL MOOC module on Handling Large-Scale Data Using Stata. We are here to explain the unit level data and their prerequisites. Within that, we have been trying over last two lectures on tabulation and creation of new variables. So, what we did in the last class I will continue accordingly. One important point I feel that it might have been missed or it was there, but we did not emphasize much. I personally feel that you guys might be in trouble in dealing with the gen and replace command. So, I am starting with the recoding or the generate and replace variables once again.

I already mentioned that when you have given any kind of if condition to generate and replace, one difference between generate and replace is that, generate, when the if condition is attached, if condition means you have limited your observations or the variable with the particular more information and with some mathematical operation you have limited with less than 2, if it is there, you just simply keep, others you do not keep. So, those will be converted to others, if you have defined less than 2, others will be converted as missing value.

(Refer Slide Time: 02:07)





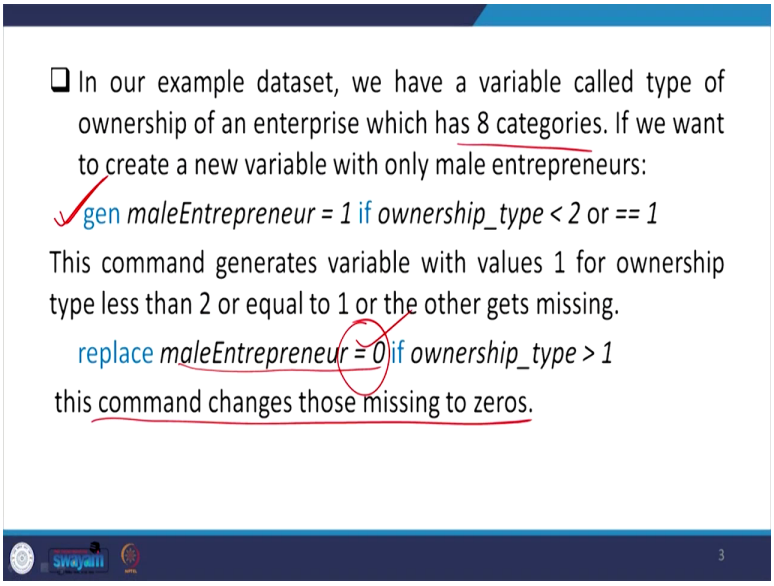
Whereas in case of replace, those will, whatever is there basically it replace the values, but it will not convert others to be as missing. So, that is the only difference between replace and generate. Now, you might be intriguing to know what is the best one between generate and replace. I think, that way it is not defined as best or worst or bad or good, generate is useful to define another most important, another variable where replace is required when you know that particularly some particular entries are problematic, you need to correct that particular entry. But in case of generate, generally, it is for new variable with certain if conditions. So, both are useful. It is not like which is good or which is bad that is not the way we are explaining here.

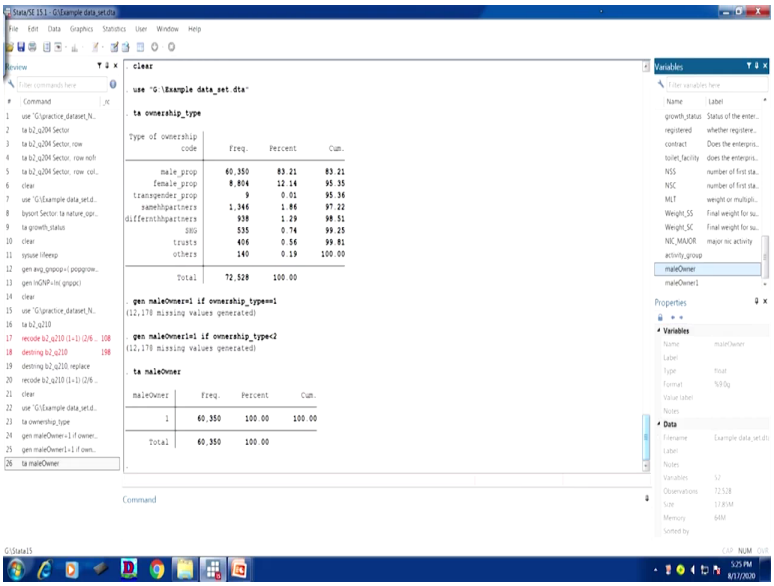
Now in our example dataset, I will straight away open the example dataset, I will start with this. I did not operate in the last lecture. So, I will start with the clear of this existing data, because there might be some overlapping possibility. So, let us start with the example dataset of ours. We will provide you to operate. How you can go for it and how you can able to experiment. So, we have an example dataset and we will also provide you and we filtered this data for you. And we are going to tell you in a short while. Why this is different than that of this, because we have already recoded, we have destring some of the variables for better use.

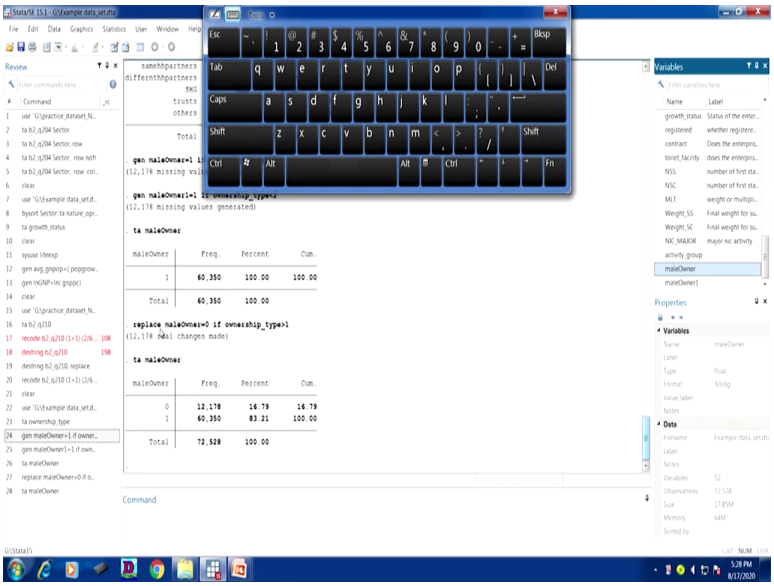
And destring I think I said that we are going to continue. We have already explained destring from string to numeric data. We have already mentioned earlier. Regarding string variable we have already clarified. So please do not get confused with destringing. So, we operated

destringing in the last entry. So, here like destringing we did this, so like this here. So, do not get confused. We already operated this in the previous lectures.

(Refer Slide Time: 04:29)







So, what I will try to mention here for you that if we are going to generate male entrepreneur, like for the example dataset let me first have an understanding of the example dataset, what is that. You might be confused. Let me just read what is written and please mark carefully that whatever the text we are showing for you on screen is very very important. Do not miss any single line in between that might be putting you in trouble. So, try to digest each of the words and sentences.

So, in that particular example dataset, we have a particular variable called type of ownership that I told you in the last lecture also. So, type of ownership of an enterprise which has 8 categories. Let me show you clearly. So, type of ownership is here. We recoded already ownership type. This is the original variable. Now we have already redefine it with our better use. Type of ownership is there. So, let me check what are there within the type of ownership that we have already shown you, but let me check once again.

There are 1, 2, 3, 4, 5, 6, 7, 8, 8 codes. So, there are eight codes. Here it is mentioned 8 categories. 8 codes I refer 8 categories. Different codes are entered, like starting with 1, 2, 1 code is for male proprietary, female proprietary, transgender proprietary, same household partners, likewise others are given. Now what I do, if we want to create a new variable with only male entrepreneurs, others are not important for us, likewise we filtered the data for you. We have filtered a small set of data for our own use. So, you are only interested for male entrepreneurs and other entrepreneurs are not at all important for you.

And you know very well that male entrepreneurs are having with the data 1. This is the code entered as 1. Others are different codes. So, basically what I will do, others like 2, 3 till 8 should be converted. So, like you defined the new variable as male entrepreneur and equal to 1 if ownership type is less than 2 or you can enter double equal to 1, whichever way you do that will give you the 1 as a result.

Likewise, this, so let me generate a variable generate male owner, you can simply write down male owner or like you can go by this owner, male owner equal to, now what I will do, so your condition is, but you wanted to keep male owner equal to 1, you have to stick to ownership only, so if that ownership either, there are two ways of doing it. We know that there are 8 categories.

But you are interested to keep only one or generate a variable that is having with only code 1 and others are not important to you. So, let us start with double equal to 1. What it converts. It converts with, if you enter it, it has defined. Now look at very carefully, 12,778 cases are now missing or generated. Why is it so? Not necessarily these are missing from the data, but for us, for our conditional variable it is missing. Like it has converted all other as missing at this moment, but later on if you want to convert further and decode it that we will tell in due course of time.

Another approach of doing this is, basically instead of double equal to 1, you can do less than 2 or but in this case you have to change the variable name, new variable name has to be changed. So, now suppose I was just wanted to show you, if you wanted to make it 1 like this, a new variable as 1, now instead of double equal to 1, you are interested to do it at 1, but you know that either you enter as less than 2, less than 2 means it certainly 1, so it will result in a new variable with the same information, same result. Now look at the missing value, 12,178 missing values generates.

So, this has resulted in new variable. Now you can check that new variable. Look at this, male owner or male owner 1. 1 here we have entered a new command, a different command with the same result. Now I just wanted to check with anyone of it. Now look at only 1 is there that is male owner information is there, rest are missing. Now how 12,178 is resulted, look at this. 8,804, this is female plus 9 plus 1,346, 938, 535, 406, 140, if you add these all together that will be of 12,178. Now this is we have just explained. So, now I think in the last class I did not do it. I

thought you guys might be in trouble in doing that. So, this command generate variable with values 1 for ownership type less than 2 or equal to 1 or the other gets missing.

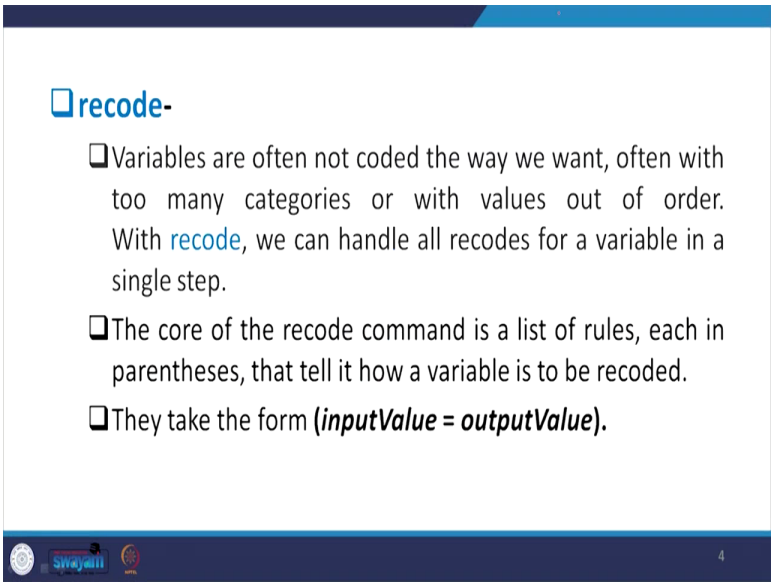
Similarly, in case of replace, if you do it, replace male owner equal to 0 if ownership type is greater than 1. What do you mean by that? Like you wanted to make male owner to be replace with 0 if ownership is greater than 1. So, ownership type if it is greater than 1 that means if any other like 2, 3 till 8, if you just enter this command, the way I have entered generate male equal to 1 if ownership less than 2 exactly if you just copy and paste and greater than 1 you will find the same one I think.

This is important to note here that this command changes those missing to 0. The missing is converted to 0, because in our command we have replaced a 0, isn't it. In our command we have replaced a 0, so like I can do that, but since this I have already done it, so replace in the place of this we will only use replace. Replace or what you do instead of typing so many times you simply click it, click here. So, now in the place of this you do like this, replace. Now ownership type replace, but you here replace male owner equal to 1 if ownership type is double equal to, but in this case it is not double equal to 1, you need to understand if it is greater than 1. You wanted to convert others to be zero. You wanted to make this to be zero.

Suppose we are just trying to clarify how it works. Like others you are not interested you simply want to replace it to greater than equal to 1. If it is greater than 1, that means it reads all other entries with replace male equal to 0 if ownership type is greater than 1. If we do that, now it has replaced. Now look at the same number 12,178 has been replaced with 0. Now if I check it, tab male owner, look at this. Now male owner 0 is of how many, 12,178 which we have replaced with 0. We have already mentioned that. It has already resulted 12,178 replace have been made.

Now you just mark another thing important that it is not converting to missing values. So, in case of generate, if it does not suit your result within that conditioning command, if it is not within that preview then it will convert to missing. But here it is converting to 0 because we have required to replace it with 0, others are intact, isn't it?

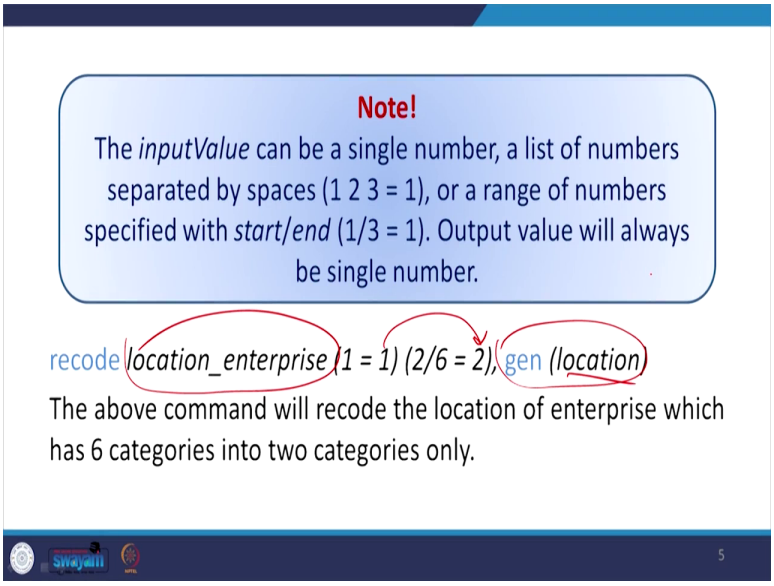
(Refer Slide Time: 14:54)



So, let us start with our other details, so recoding. Recoding, I already discussed a bit in the last class, but I will clarify further. Variables are often not coded the way we want, often with too many categories like, in our case also, ownership type there are 1 to 8. We are not interested to look at all 8 together. Suppose we wanted to male versus others.

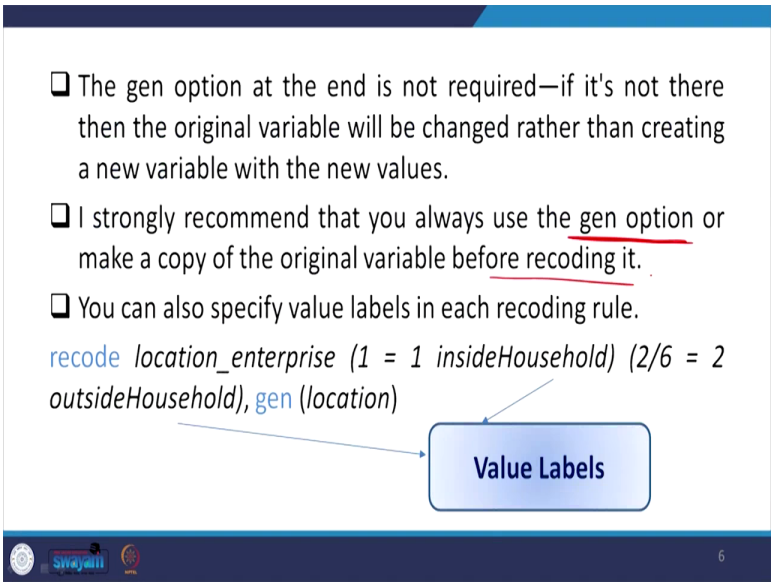
So, we can make it others, how other counterpart is also competing with male. If you wanted to just categorize 2 to 8 we can recode it too, recode those 2 to 8 to another value, maybe 2 instead of all other numbers. So, we will tell you. The core of recode command is a list of rules in each parenthesis that tell it how a variable is to be recoded. They usually take the form input to output value.

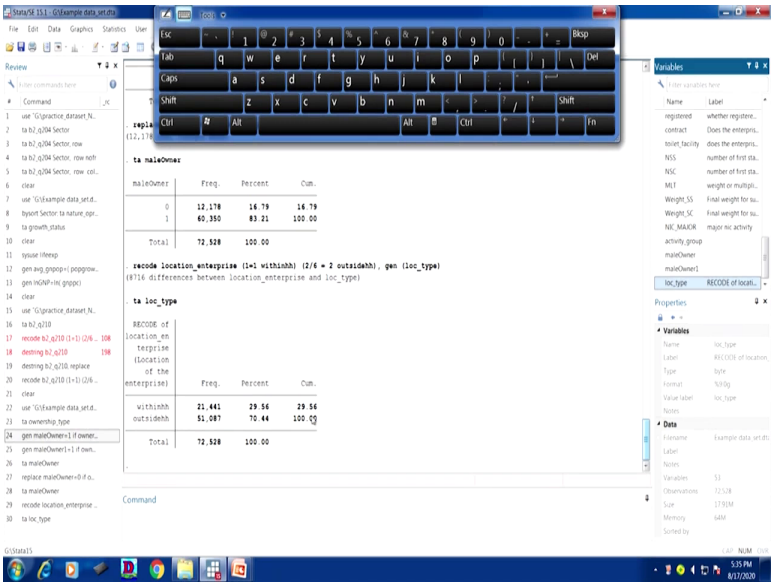
(Refer Slide Time: 15:52)



And here the recoding I already told you that if you recode it the location of the enterprise or even the same enterprise, I already shown you in the last lecture that I defined a variable with loc so please mark carefully and you please recode and go through accordingly.

(Refer Slide Time: 16:17)





I think I should not spend much time. There are important interpretations to be covered. The generate option at the end is not required if it is not there, then the original variable will be changed. Basically, if you are not adding with a generate, generate here like this, if generate is not there, what we will do, basically it will change the original variable. But it might be the case that you want the same variable again in different set up or different revised format, so you cannot revise further, because you have already generated with this specified code.

Now, your new variable location will be available with 1 and 2, only 1 and 2 will be available. In between codes like 3, 4, 5, 6 would not be available. So, it might be difficult for you to work in future. So, now if it is not there, then the original variable will be changed rather than creating a new variable with the new values. I strongly recommend that you always use the generate option. It must be used. And make a copy of the original variable before recoding. So, generate option make a copy automatically if you add comma gen.

So, you can also specify value labels in each recording rule. Recode like in the generate also like recode location we did here, we did like recode enterprise 1 equal to 1 and location of the enterprise 2 to 6 as 2. What are these 1 and 2 now as per the new variable we generated. So, in this case 1 equal to 1 if you do it. You simply mentioned that within the household or inside the household or 2 to 6 equal to 2 as outside the household. Let us operate this. I think it is there. The variable name is our location of the enterprise, here location of the enterprise.

So, we are going to operate location of the enterprise but now we are generating with a new variable. So, what we do, we are interested in recoding with a label. So, let me recode. I recode the location of the enterprise. But make sure that you have checked the location of the enterprise carefully. Let me check whether a new variable, it is not there. So, I will start with the location of the enterprise here.

Now the way I did it 1 equal to 1, but you have to, within the bracket we have to, likewise this, we have to specify the way it is there you should not miss anything otherwise it will not read as the label. You have to specify in addition to this. So, let me stick to. So, here location of the enterprise space, now what I did here, 1 equal to 1, but 1 equal to 1 is what, let it be within household, inside the household or within household whatever you write, within hh, then you close the bracket.

Then another one you wanted to recode and also you have given a label here. There are other approaches of labeling as well. We will also tell you in due course of time. But now 2 basically there are till 6, 2 to 6 is equal to 2, 2 what is it, outside hh. Now you close the bracket. Now since you have recoded it, we suggest always that you generate a new variable, generate command with a new variable within bracket you can take the name location underscore type. If you do that, it will generate you the loc type. Now, loc type has already been defined for you.

Now, you want to check what is that, you just check that. You will get the information. Within household how much, what is the frequency and outside the household how much is the frequency is clearly derived. Now, so we have so far learned value labels also with the same recoding approach. There are labeling of the variable. This is value labeling. We will also discuss about variable labeling and the code labeling and how value is defined we will also discuss in our lecture.

(Refer Slide Time: 21:57)

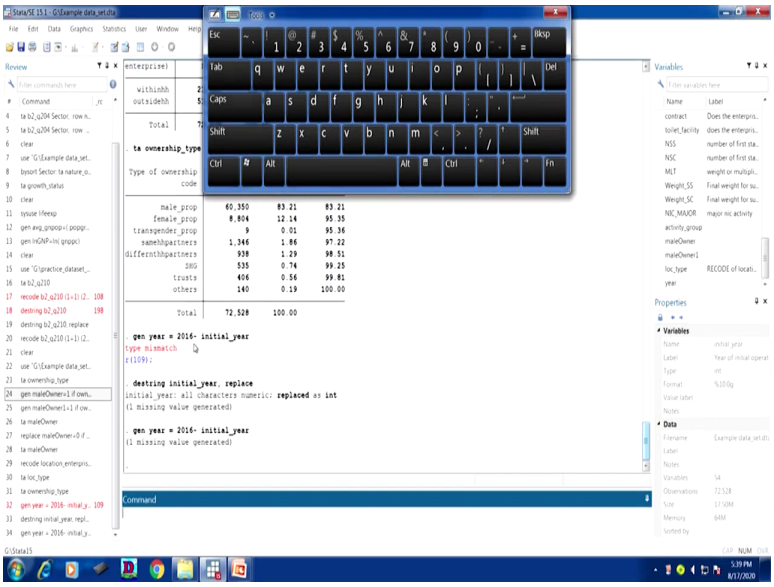
egen-

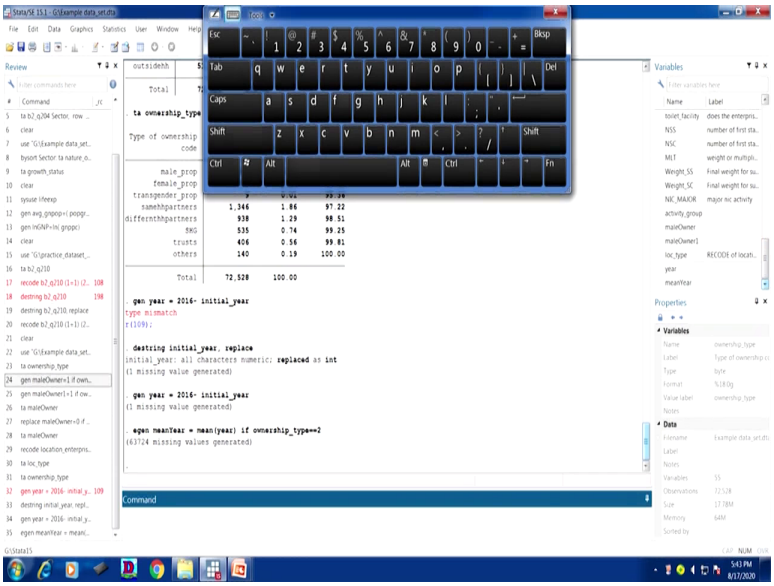
- The egen command, short for "**extended generate**" gives you access to another library of functions.
- The egen command typically creates new variables based on summary measures, such as sum, mean, min and max.
- Suppose we are interested in knowing the average year of operation of women entrepreneurs:

  egen *mean_year* = mean(*year*) if *ownership_type* == 2

Note: here, category 2 consist up female owned enterprises.

- Egen is wonderful for saving summary statistics as new variables.





In addition to that, there are some egen command also. Am I missing something, no? So, I think I have covered. So, egen command is very interesting to learn because it gives certain different approaches of understanding the variable generation. Egen command is short form of the extended generate, gives you access to another library of functions. The egen command typically creates new variables based on summary measures such as sum, we did it, mean values, mean, maximum.

Suppose we are interested in knowing the average year of operation of women entrepreneurs, average years, instead of, it seems that mathematical operation is already involved in our objective function. So, our objective function is to know the average year of operation of women entrepreneurs. In that case, simply ownership type, in case of average years, what I do, egen command is very important. Egen then you define average years as mean year is equal to mean within bracket year if ownership type double equal to 2.

Now, let me first check ownership type, I just want to check. So, ownership type, but it has different types. Average years we wanted to know, years of operation. So, there are the variables we need to define with that year based on the data. We need to stick to that data first. Since the variable is already in string format, so becoming difficult to replace. So, now after destringing we can able to do that. So, the year is defined here, year as the variable.

So, accordingly now the way we did it like generating year 2016 minus initial years so the number of years will be calculated, because it has given the entry as before 2016, 2015, 2014,

2013, so number of years you can do it if you generate as a year variable with subtracting the exact year. So it will subtract it and define the number of years of operation, isn't it?

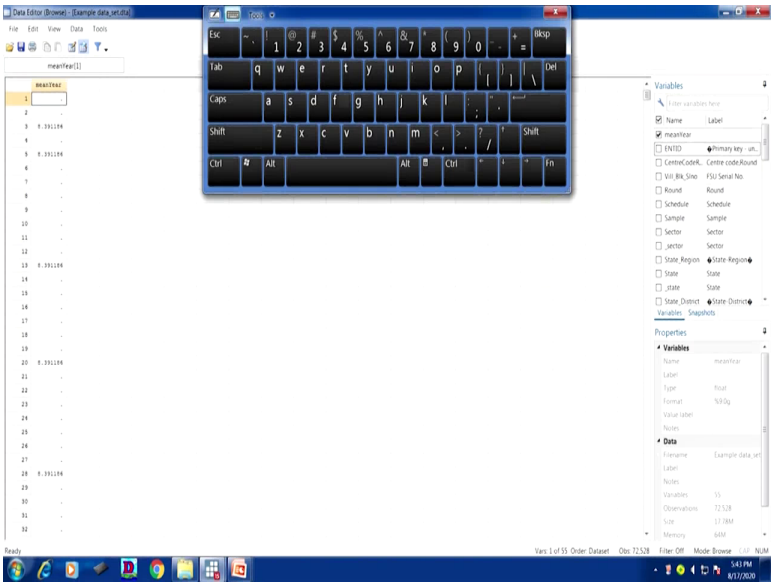
So, now once years of operation is defined, you wanted to know average years of operation of women. So, how to do it average years of operations, so basically egen command is going to be very useful. Now, in that case simply mean year if condition, if it is there you will do it, by ownership type if you do it, it will convert it to the variable. Otherwise, if you do not have any if condition, it will only give you mean of years. So, you can go through accordingly.

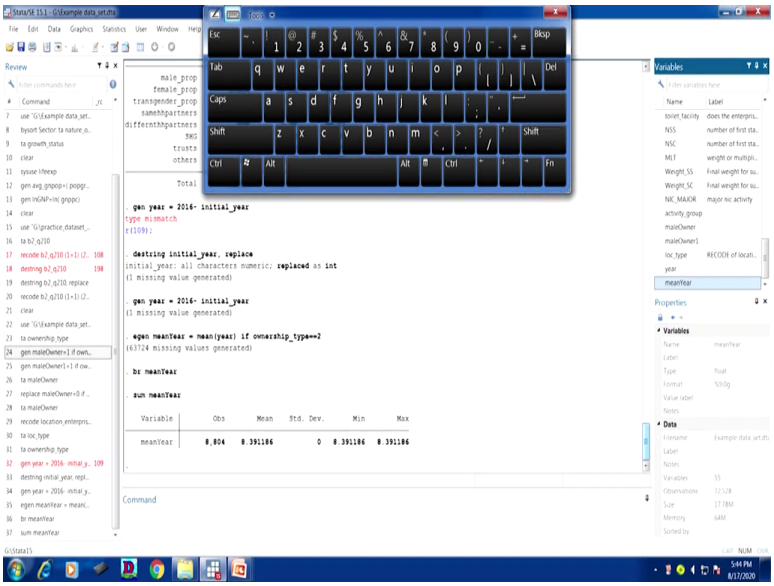
Now, please note carefully that the category 2 consist of female owned enterprises. Now, so what we derived here based on our average years of operation of women, our requirement is for women. If it is not women then ownership type is not required to define. Since it is ownership type, I think I already told you, there are 8 type of owners. So, 1 is for male and 2 is for female proprietorship. So, you have to go by a conditioning of if, since our objective function is on women.

If it is only average years of operation, then you need not require the if condition. So, like egen, we can do that. So, egen command if you do it like egen, you have to define a variable mean year. The way we did it, we have to do by the same name, mean underscore year, mean with this, is this fine, is equal to, so in that case you have to take the year variable. So, in that mean, isn't it. So, likewise this, here mean but I think Stata reads mean. Mean is the year for the average. So, mean then within bracket the variable we have already defined for year needs to be attached or you simply click on that variable it will automatically read then.

Now this is important because till this, it only gives mean years of operation of the entire enterprises if I just go by this. But since our objective function is for women, I have to condition a variable if condition must have been there. So, if conditioning must be there, but I think it may not read the comma, because, I need to close that, so if the type of ownership I already discussed, type of ownership here should be double equal to, here double equal to function is very important, because you need to specify the exact requirement or exact conditioning then it will convert it into the mean year. You can check it here; the mean year is already being defined.

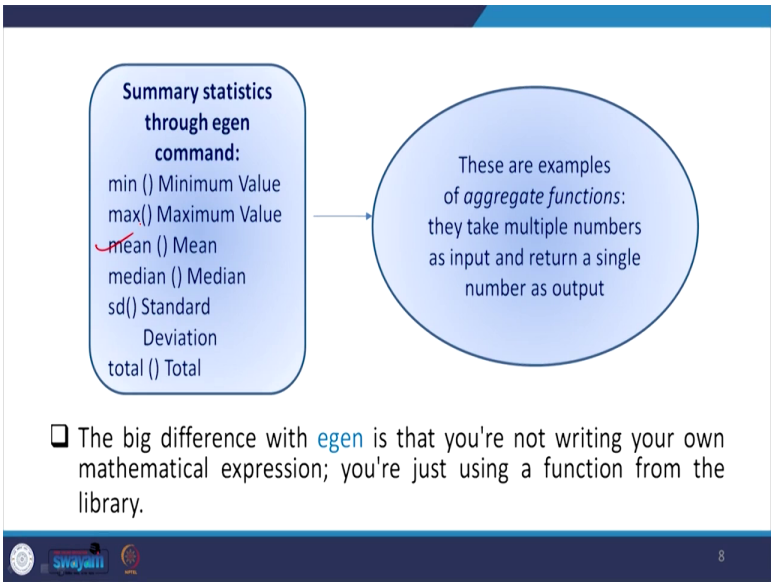
(Refer Slide Time: 29:04)





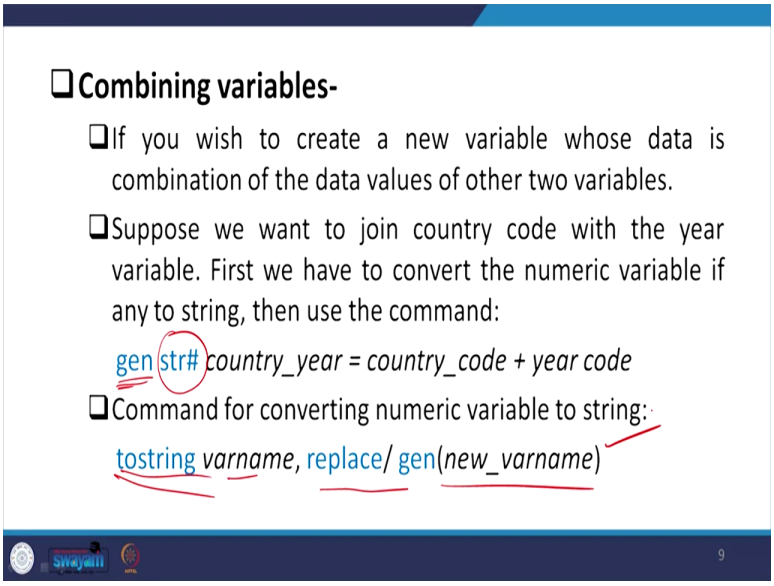
Now you can check through this. It will give you the mean year of an operation. So, summary statistics of mean year we can also derive. We can also derive the summary statistics through sum as I told you already. Sum of the mean year it gives you the details of that particular variable, number of observation, mean year of operation of the entire female owned enterprises are of 8 plus years. Minimum and maximum values you can also derive accordingly. So, I think it is very wonderful for getting the summary statistics through the egen operation.

(Refer Slide Time: 29:47)



Now what are also interesting to note through egen is like minimum value, maximum value with mean within bracket, max with bracket, mean value we already operated, then median value, then standard deviation or the total, these are some examples of aggregate function. They can take multiple numbers as input and return a single number as output. The big difference with egen is that you are not writing your own mathematical expression. You are just using the function from the library from the beginning of your operation and at the end you will get the correct result.

(Refer Slide Time: 30:37)



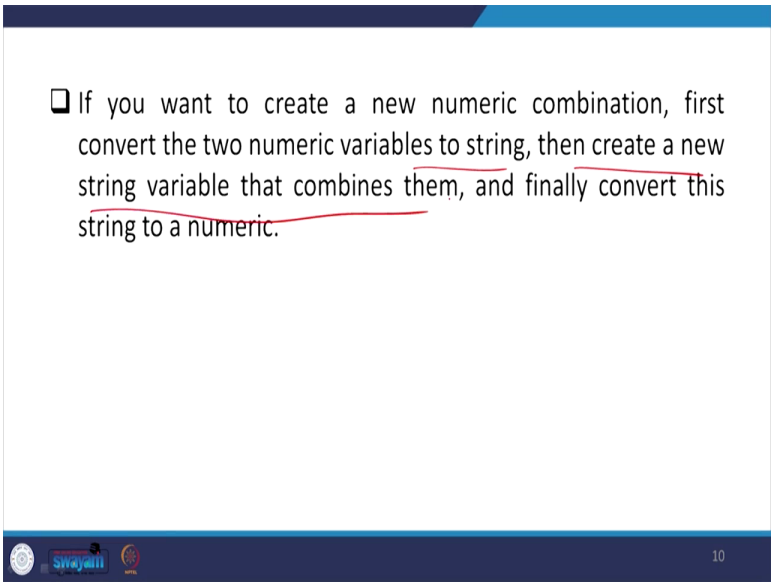
Now also it is important to know, combining variables. So, far we discussed generating variable with replacing also the codes, also we discussed labeling the codes through generating a variable. Now combining variables, there are some technicalities. All variables cannot just combine on its own, it depends upon whether it is string or numeric. You need to be very careful. How to check we already told you.

If you wish to create a new variable, whose data is combination of data values of other two variables, but it has to be added with another two variables. For example, we have taken two variables and its average. But here suppose you wanted to combine two variables like we want to join country code with the year of that variable, country code and year.

As you know that these two cannot be just combined, first we have to convert the numeric variable, if any, to string then use the command. That is here it is given generate, the way we generated command is already given, but str must have been given there, str command, string hash country string then number then country, year is equal to country code plus year code. Then in that case, since the variable is in string it will then only the country, basically, country number and year you wanted to add together so you need to give that country number here. Otherwise, it will not just add.

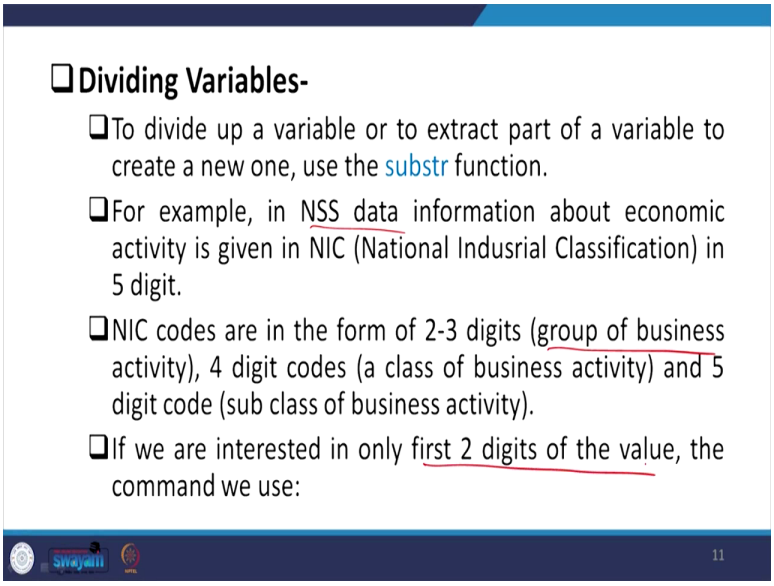
Similarly, command for converting numeric variable to string then in that case another command I think we have not discussed earlier. Here we are discussing tostring. tostring then variable name to be added, replace or generate. If you wanted to generate a new variable then you have to give the variable name the way you did. If it is string variable then tostring variable name replace you can accordingly do it, but depending upon how you are converting it to. If you are converting into string then you have to do tostring, numeric variable to string then only tostring is important. But if it is string is there, you wanted to get it in numeric then destring is important. So, then only you can able to combine.

(Refer Slide Time: 33:33)



Now let me understand if you want to create a new numeric combination, first convert the two numeric variable to string, then create a new string variable that combines them and finally convert this string to numeric. We have already mentioned.

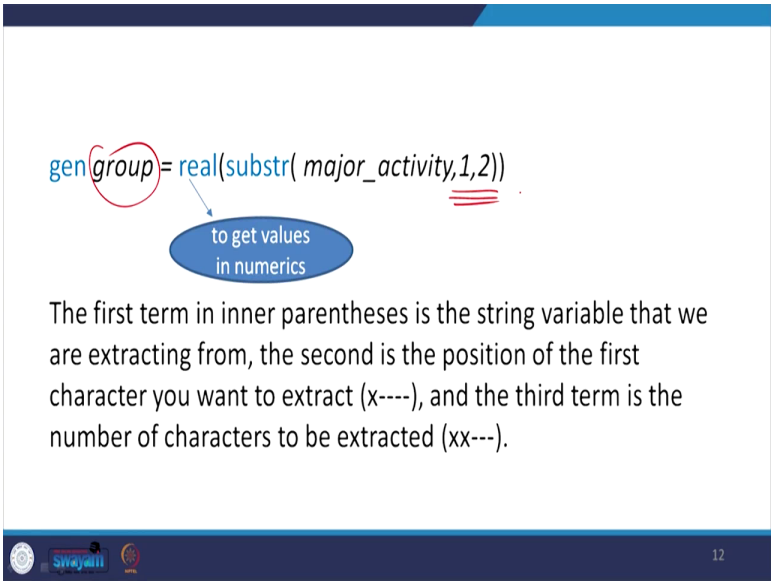
(Refer Slide Time: 33:53)



Now, similarly, these are some important aspects while we operate. Dividing variables is also important. To divide a variable or to extract part of a variable to create a new one use the substr function, sub-string function. For example, in NSS data we can open, NSS data information about economic activity is given in NIC classification that is of 5 digits, 5-digit classification.

The NIC codes are in the form of 2 to 3 digits, that is basically, groups of business activities coded with 2 to 3-digit space and 4 digit codes, a class of business activities and 5-digit code, but those are of sub-class of those business class activities. If we are interested in only first two digits, two digits of the value, the command we are going to use is like this.

(Refer Slide Time: 35:04)



Generate the group with first two digits, the name of that, we are interested in only first two digits. So, generate basically group we mean we are referring to a new name with first two digits, but first, every time I told you here whatever you are writing after generate or anything this should be a name of that particular variable. And then you have to add the real number that is to get the values of the numeric number of sub-string of that particular variable we wanted to convert that is the major activities which has 1, 2 till 5 digit NIC classification in NSS.

We will also show it in our next class in detail NSS data. It is there already, but we could not have opened it. But since is unnecessarily consuming more time, we will use it for sure in the next class in detail. But let me tell you what it defines. So, it defines with the two-digit classification of the NIC codes. The first term in the inner parenthesis is the string variable that we are extracting from and the second is the position of the first character you want to extract, and the third term, likewise this only, the third term is the number of character to be extracted. I think, are there any further doubts we will clarify further and also repeat some of the things in the next class. With this, let me close here. Thank you.