

So far we have explored the user interface, looked at a dataset, and navigated to the folder where we want to work. At this point, I would like to introduce you to a very useful command. We have actually already seen this command in a previous lecture, but now it's time to see how we can get help regarding a specific topic. So far we have used several commands, most notably the **use** command, which is used in order to load a dataset. If you want to know more about this command, you can type the following:

help use

A new window pops up. This window gives us a lot of information about the **use** command. The section division is standardized. So whenever you ask Stata about help regarding any command, you will get a window with the sections syntax, menu, description, links to PDF documentations, options, and examples. This will make it easier to get the information that you are looking for. Stata also comes with PDF documentation which can be accessed by clicking on the link shown at the top. The PDF documentation always contains more information and examples about each command and it is an extremely useful resource.

The syntax section tells us about how the command is written. We can see that to execute the command, we have to type **use** followed by the file name. Stata also tells us that the command has options. In Stata, most commands have options, and these options are written after a “,”. There are two options available with this command, and they are **clear** and **nolabel**. If you want to know what each of these options does, scroll down to the options section. We can see that the clear option tells Stata that it is ok to replace the data which is in memory even if the data is not saved. You need to specify this option if you are trying to load a dataset and you have another dataset that is currently loaded. Stata will not allow you to replace the current dataset if you have not saved your work. If you don't really care about the changes that you have made the dataset that is currently in memory, specifying the **clear** option tells Stata that it is ok to replace the unsaved dataset and to load the new dataset.

The other option, which is **nolabel** specifies that you want to use the dataset but do not want to load the value labels. We have still not covered what a value label is so you don't need to worry about this at the moment. What I want you to know is that if you want to load a dataset without loading the value label, you will have to run the command:

Use "dataset name", nolabel

The help file also includes examples to show us how to use the command. The first example is actually interesting because it shows us that we can use a dataset that is stored online and not on your computer. This is a very nice option. For example, if you have an internet connection, try running the following command:

use <http://www.stata-press.com/data/r15/auto>

The Stata output tells us that a dataset has been loaded and that the dataset is labelled as “1978 Automobile Data”. Looking at the right-hand side we can see the variable names. If you run the

command **describe** you see that the dataset has 74 observations, and you see a list of the variables with their labels.

We can save a local copy of this dataset by using the command **save**:

```
save myauto
```

Stata saves the dataset to the current working directory. If we now run the **ls** command we can see that there is a new file named *myauto.dta* in the current folder. Since we have just used a new command, let's ask Stata for some information about it by running the command:

```
help save
```

We can see that this command is used to save the data which is currently loaded in the memory to the file with the file name “filename” which was *myauto* in our case.

As you can see, Stata's commands are intuitive. It is also very easy to get help when we need it. The help file which opens up in the new window tells us about the syntax, the options, and gives us some examples. If we want more details, we can click on the PDF manual entry on the top of the window.

The help command is useful when we need help regarding a certain command, but what if I need help regarding a certain topic? For example, what if I wanted to know what type of tests would I use to test the normality of the data? It would be useful if Stata had something like a Google search engine. Fortunately it does, and we can access it using the **search** command. Type the following:

```
search normality test
```

A new window pops up. The window contains many entries. At the topic, we are told that the first group of entries were found in the official help files which are already stored in your version of Stata, FAQs, examples, SJs, and STBs. SJ stands for Stata Journal while STB stands for Stata Technical Bulletin. Stata has an official journal which published new articles about recent developments in the program, especially with regards to the new commands. The **search** command searches in all of these resources. If we move down the window a bit, we see that there is another group of results under the heading “web resources from Stata and other users”. The **search** command not only gives us results from official Stata publications and official Stata help files, it also returns search results from user websites. As you can see it is a very good resource.

Going back up to the search results in the official category, we can see that there is an entry called the “sktest” which is described as “skewness and kurtosis test for normality”. This sounds like what we are looking for. The search result tells us that this command is already installed in our version of Stata and that it has its own help file. We can either go to the command prompt and run the command **help sktest** or we can simply click on the blue link. Clicking on this link opens the help file for this command. The help file contains the usual information. Don't worry at this point what this command is or what does it do. This is not the purpose of this exercise. The purpose is to let you know that the search command in Stata is very helpful and that it returns results from a wide range of resources.

What if we wanted to check out one of the results that was not already installed on our computer? If we run the search command again, we can notice that some of the search results start with the letters SJ. As you can guess, Stata is telling us that this result is from the Stata Journal. Stata also tells us which volume and issue, and even the page numbers of the article. Clicking on the blue link will take us to a page which tells us about the authors of the command and that also contains a description of the command. Stata also provides us with a link that we can use to install the command onto our computer. If you want to install this command, you can just click on the link and Stata will do the rest. You don't have to worry if you had previously installed the file or not. If it has been previously installed Stata will tell you, just like in my case. Otherwise, Stata will install the command. If we go back to the previous screen, we can see that Stata tells us the name of the help file for the command, which in our case is **chens**. Go back to the command prompt and run the command:

help chens

You can see that the command, along with its help file, has been installed. As you can see, installing new commands in Stata is really easy.