Laboratory Tutorial 1-8: SPSS Command Syntax

In this laboratory tutorial you will:

1. Learn how to extract SPSS Command Syntax for the functions that you use
2. Learn how to edit and run command statements in the Syntax Editor

This is not a mandatory tutorial but you are strongly advised to complete it as it will help you to pass the coursework.

Preamble

In this module, you have learned about statistics using a package called SPSS. SPSS is a powerful application but, so far, you have only accessed this functionality through the graphical user interface (GUI).

One particularly useful feature of SPSS is the facility to access commands using written statements. This can be useful if you need to run a particular command or sequence of operations more than once. SPSS command syntax is a type of programming language. It is similar, in principle, to SQL in that it is a declarative language – statements describe what operations should be performed, rather than how they should be performed. The underlying SPSS 'engine' translates these statements into the relevant computations and outputs.

It is possible to combine command statements into ordered sequences or scripts. For instance, you might want to create the same set of statistics for more than one dataset, monitor the change in statistical outcomes as a dataset grows, or perhaps explore the effects of different transformations or test configurations on one or more statistical tests. Using syntax in this way is much more efficient than revisiting the GUI dialogues each time.

For the coursework (in semester 2) , you will use SPSS command syntax as the way to communicate how you attempted to answer questions using SPSS. This will apply principally to the Higher Questions. When answering these questions you will be asked to describe your method as well as the actual answer. If you used SPSS to arrive at your answer, then paste the command(s) that you used into this field. Make sure that you paste multiple commands in the correct order in which they were executed (run).

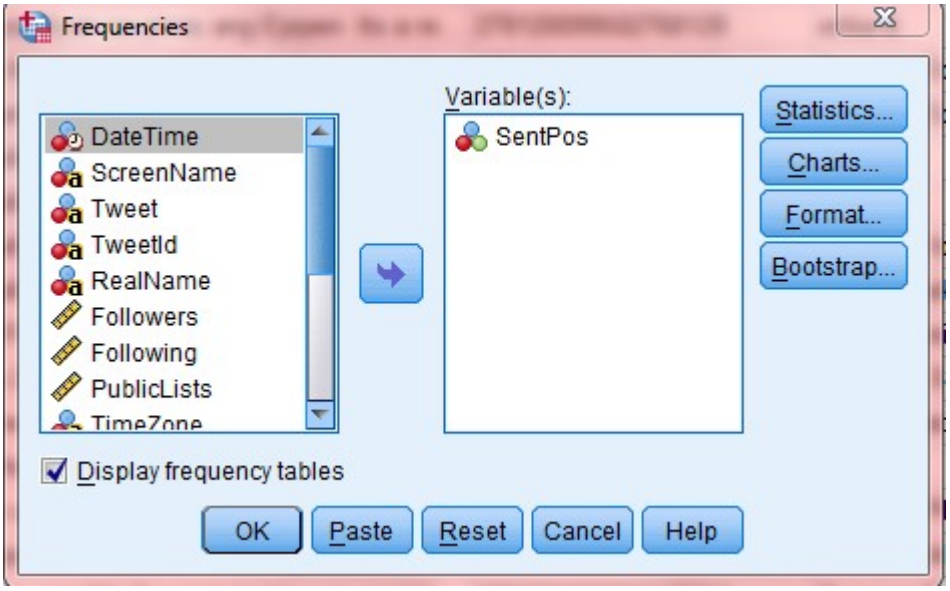
In the following exercises, you will be using the **EpipenTweets.sav** file, which can be found in the resources folder. *Epipen is the trade name for a type of injection device used by people who are prone to severe allergic reactions, known as anaphylaxis, to certain agents like bee stings and peanuts. Epipen injects adrenaline into the blood stream, which can provide immediate relief from life threatening symptoms such as a constrained airway. Hence, for sufferers, an Epipen can be a life-saver and must be carried with them at all times. What can this Twitter time line tell us about users of this device and their experiences living with their condition?*

Exercise 1: Extracting commands from the GUI or Output Viewer

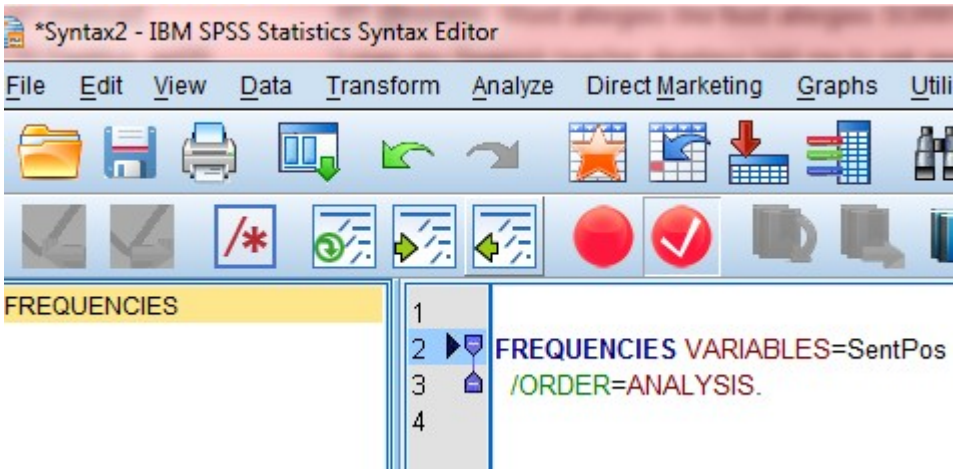
There are three ways you can generate commands. The first is to type them by hand. Of course this requires some working knowledge of the command language. For beginners, it is easier to work using a copy/paste/edit approach. There are two ways to do this. The first is to paste commands from the function configuration dialogue. The second is to copy and paste the command from the Output window. We'll look at the former method first in this section.

You may have noticed when configuring a function (e.g. Recode, Descriptives, t-test etc.) that next to the **OK** button on the dialogue is another button labelled **Paste**. If you click this button it will paste the command syntax, corresponding to the configuration you have set, into a **Syntax Editor** window. If you don't have a Syntax Editor open, one will open automatically.

Let's try a simple example using the Frequencies function. Suppose we want to compute the frequency distribution of positive sentiment scores across Tweets. Go to Analyze → Descriptive Statistics → Frequencies to bring up the Frequencies dialogue and move SentPos into the Variables list as shown below.



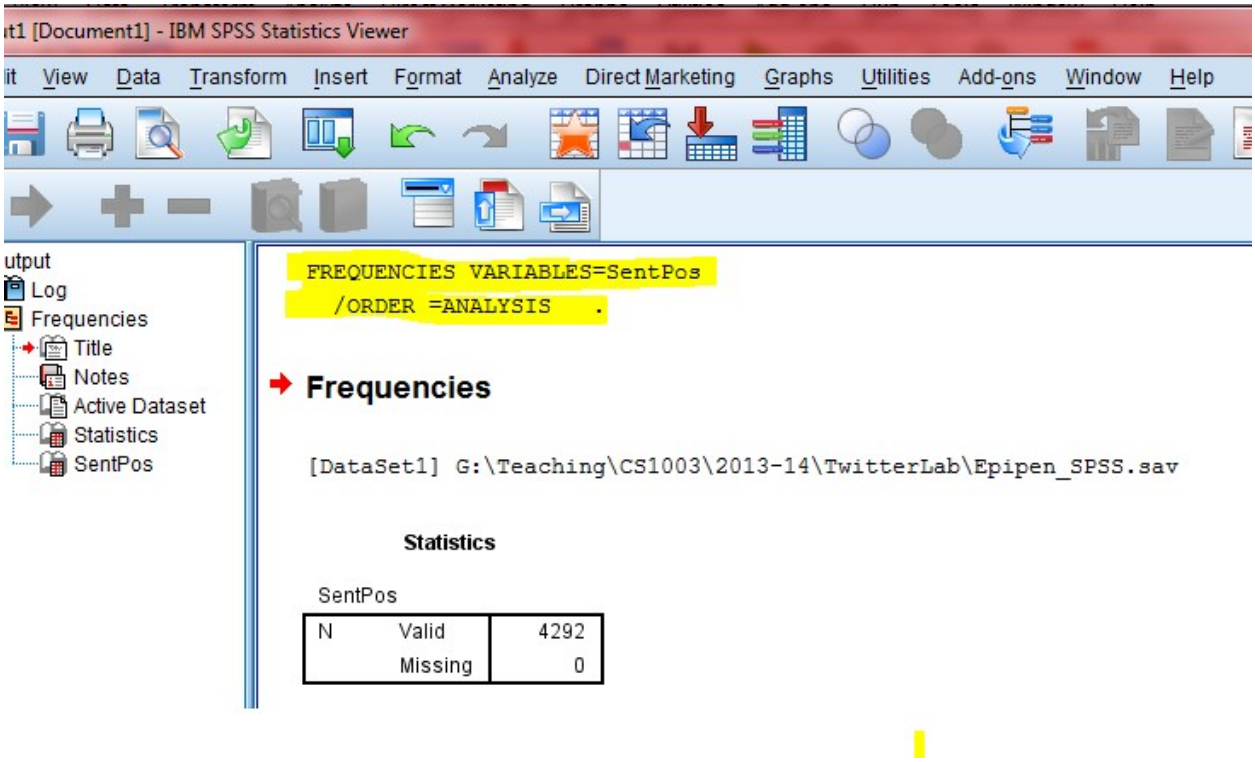
Let's run this command. Now, instead of clicking **OK**, as you have done before, click the **Paste** button. You should find that a Syntax Editor window opens and that it contains some text as shown below:



Take a moment to examine the syntax statement. We can see the following keywords/clauses:

- FREQUENCIES – this is the name of the function/operation that we wanted to call
- VARIABLES – this keyword prefixes the items in the Variables list. In our case it's SentPos, but now that we are in the editor, we could edit this to any valid variable name. We could also add additional items, simply by typing their names, separated by a space (e.g. 'SentPos SentNeg RealName')
- /ORDER – this specifies the order in which results are output. This is only relevant if you have multiple variables in the list. The default is ANALYSIS which means that each analysis is presented once, with the results for each variable aggregated together. The alternative is VARIABLE which would deal with one variable at a time, outputting the results for each analysis, before repeating the output for the next variable.
- The statement terminates with a full-stop.

All you have done at this point, is pasted the command syntax into the editor. In order to execute the command you need to select the statement and click the green 'play' arrow. The results of the operation will appear within the output window, just as they would have done had you clicked OK from the dialogue.



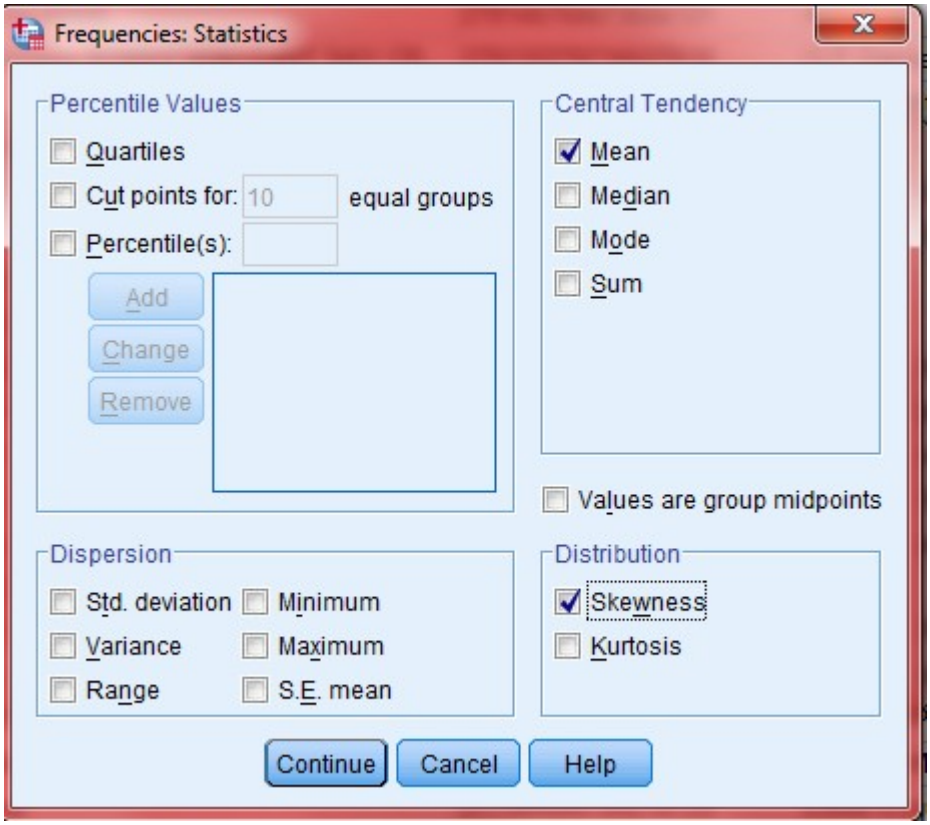
If you look at the output that appears, you'll notice that the statement that you just executed is also displayed prior to the results. Hence this is another, but less tidy, way of accessing syntax commands i.e. you could manually copy and paste this text into the editor.

Exercise 2: Editing and running commands from the Syntax Editor

The above command statement example is quite minimal. We just entered one variable and made no other changes. However, most functions in SPSS have a range of different, optional, parameters which can be accessed and set using keywords. Some of these are accessible from the GUI but others can only be accessed using typed commands. Hence, learning command syntax provides you with full access to and control over the more powerful features and functionality of SPSS.

Let's take another look at our **Frequencies** statement. If you revisit the GUI dialogue, you'll notice some buttons on the right hand side. Let's start by clicking on **Statistics**. This allows you to select from a range of averages, dispersion measures and other statistics. Let's see what happens to the Frequencies statement if we select one of these options.

Select **Mean** from the **Central Tendencies** frame and **Skewness** from the **Distribution** frame as shown below. Click **Continue** to go back to the main dialogue. Then click **Paste** and go back to the Syntax Editor.



You should see a new section of command syntax below your earlier statement. Take a moment to examine the differences. You'll see that there is one extra line beginning with the keyword /STATISTICS=. This keyword is followed by a list of the statistical measures that you selected. There are actually three parameters – SPSS has assumed you want to compute the Standard Error of Skewness (SESKEW) as well as Skewness. You could delete that if you wanted to; this wouldn't of course be possible if you were just using the GUI.

```
1
2  FREQUENCIES VARIABLES=SentPos
3    /ORDER =ANALYSIS    .
4
5
6  FREQUENCIES VARIABLES=SentPos
7    /STATISTICS=MEAN SKEWNESS SESKEW
8    /ORDER=ANALYSIS.
9
```

You can add further statistics without returning to the GUI. You might be wondering how to find out what measures are available and the keywords that describe them. The editor offers a form of intelli-sense (context sensitive suggestions), similar to an IDE like Eclipse or Visual Studio. It's a little bit flaky but is helpful if you know how to use it.

Try deleting and replacing the '=' character following /STATISTICS. You'll see a menu appear with a list of keywords in it. In this case, the set of possible measures is pretty much the same as what we could access from the GUI. However, note that it is possible specify ALL measures, which saves you

from having to type every keyword into the command line. Try adding some keywords and see how this affects the output when you run the statement.

In addition to statistics there are other sub-commands accessed using keywords prefixed with a forward slash. If you create a new line (before or after the /STATISTICS line) and type a forward slash, a menu should appear showing you all the options.

Try adding sub-commands to your statement to do the following:

- Display a histogram
- Display a pie chart
- Sort the rows of the frequency table in descending order of frequency

The first two should be obvious, but the keyword for the last task is less so. Go back to the GUI if necessary and find out how this option is referenced (labelled) there. Another tip is that the first time you select a sub-command keyword, you should always type an equals character to see if there are any parameters that you can set.

Finally, you can use the inline **Help** function to access a full reference guide to the command syntax. You can search by topic or if you right click on a command statement in the syntax editor and select **Help** it should take you directly to the relevant web page.

Summary and further work

In this tutorial you have learned how to access, edit and execute SPSS command syntax statements. So far you have only focused on the Frequencies command. You are encouraged to explore the command sets for all of the functions you have learned about during the module, including t-tests, correlations and also Compute and Recode. Remember that during the final coursework task you will need to communicate all of your SPSS methods using the relevant command syntax. After making any changes, check that the command statement runs correctly on your data before you copy it into the coursework submission form. As always, remember to save any important syntax files before exiting SPSS.