CHAPTER 5

Managing Data Files

Chapter Purpose

This chapter introduces fundamental concepts of working with data files.

Chapter Goal

To provide readers with skills to read data files created with other software, and to merge data files by either adding cases or adding variables.

Chapter Glossary

Adding Cases: Merging two or more data files so that each file contributes cases, but not variables, to the new data file.

Adding Variables: Merging two or more data files so that each file contributes variables to the new data file.

Key Variable: When adding variables, the variable on which the files are sorted to ensure that data from each file, for a given record, end up properly matched.

Text File: Data saved in ASCII format using a word processing or other program.

Earlier in this book, we entered the data from the Wintergreen study using the SPSS Data Editor window. However, there are other ways to enter data, and sometimes you may wish (or need) to enter data using an alternative method. Similarly, sometimes someone else may do the data entry and provide you with the dataset, but they may not have used

SPSS for the data entry. In this chapter, we will look at two alternatives for getting data: reading data from a *text file* (also known as an ASCII file) and reading data that have been entered using another software package. This chapter will also explore a second topic of importance for managing data files, namely, how to combine two different data files (please note that for learning purposes, the topic of managing data files has been saved until this chapter, although in practice, one typically combines data files before manipulating the data).

READING ASCII DATA

Suppose that you had to enter the Wintergreen data using a computer that did not have SPSS, so that you could analyze them later using a computer that did have SPSS. To do so, you could simply use your favorite word processor. Remember to save your data as a *text file*, so that the special characters that define a *document* are not included in the file. The typical word processing program running in the Windows environment will provide you with an option to save a file as "MS-DOS text" or "Plain Text" with MS-DOS text encoding and "CR/LF" (carriage return/line feed) to end the lines. If you were entering the Wintergreen data as a text file using your word processor, you would probably want to give it a name that was parallel to the other Wintergreen files and yet also unique from them. I would suggest a name such as "Wintergreen.txt."

Let's take another look at the Wintergreen data. If you type the data using this word processing program, keeping a fixed row and column format, they will appear as shown in Figure 5.1.

```
Column Number
12345678901234567890
01      9319    12001
02      4612    00000
03      5715    11000
04      9418    22111
05      8213    21111
```

Figure 5.1 Example of Text-Based Data Entry

The respondent number has been entered in columns 1–2, the Academic Ability score has been entered in columns 4–5, Parent Education has been entered in columns 6–7, Student Motivation has been entered in column 9, Advisor Evaluation has been entered in column 10, Religious Affiliation has been entered in column 11, Gender has been entered in column 12, and Community Type has been entered in column 13. Notice that I have skipped columns 3 and 8. This was not necessary (and I could have chosen to skip different columns had I wished), but I elected to have these spaces in the dataset to make the data easier to look at. This can be handy for helping you keep your place when you are doing data entry when there are many variables for each case. For example, imagine that you are entering the data from a survey that has 5 sets of 10 items, for a total of 50 items. You might choose to leave a blank column between each set of items.

You may also notice that all of the numbers line up in each of the columns. This is because I have used a fixed font (in this example, I have used a font called Courier New). If you use a proportional font (for example, Times New Roman), your data will be much harder to see. As I mentioned earlier, this type of data file (in which all the rows and columns line up) is called a *fixed-length flat file*. Go ahead and enter these five cases, and save them in a text file called "Wintergreen.txt."

As an alternative, you may use a form of data entry that does not require the data to line up in columns. This form is known as *freefield format,* and it requires only that variables be recorded in the same order for each case and that they be separated by spaces or commas (that is, a *delimiter*). Readers interested in this format may learn more about it from the SPSS manuals.

Once you have saved this dataset as a text file, you will want to read it with SPSS. From the **File** pull-down menu, select **Read Text Data . . .** to see the **Open File** dialog box. Select the "Wintergreen.txt" data file and click the **Open** button to start the **Text Import Wizard** and view the first of six "Text Import Wizard" steps as shown in Figure 5.2.

Click the **Next** button to go to the second step. Here you want to note that the variables are aligned in fixed-width columns and that the first row does not contain variable names. The dialog box will look like Figure 5.3.

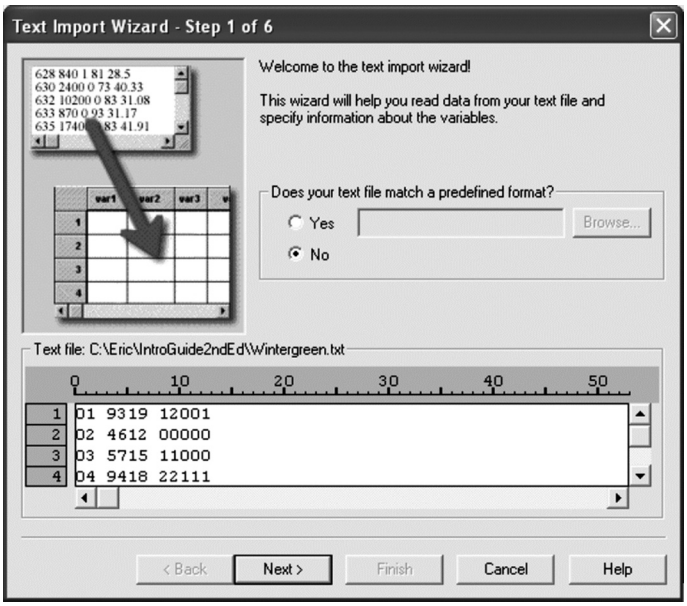

Figure 5.2 Text Import Wizard First Dialog Box

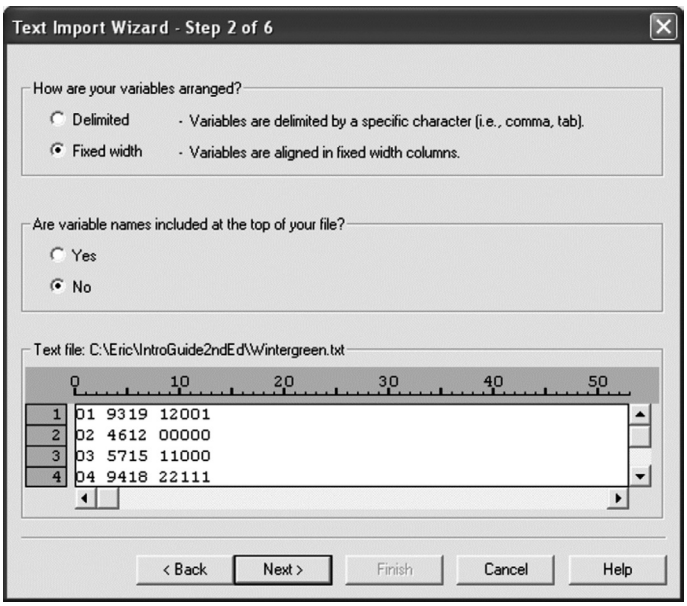

Figure 5.3 Text Import Wizard Second Dialog Box

Click the **Next** button to go to the third step. Note that the data begin on the first line, that each case requires only one line, and that we want to import all the data. The dialog box will look like Figure 5.4.

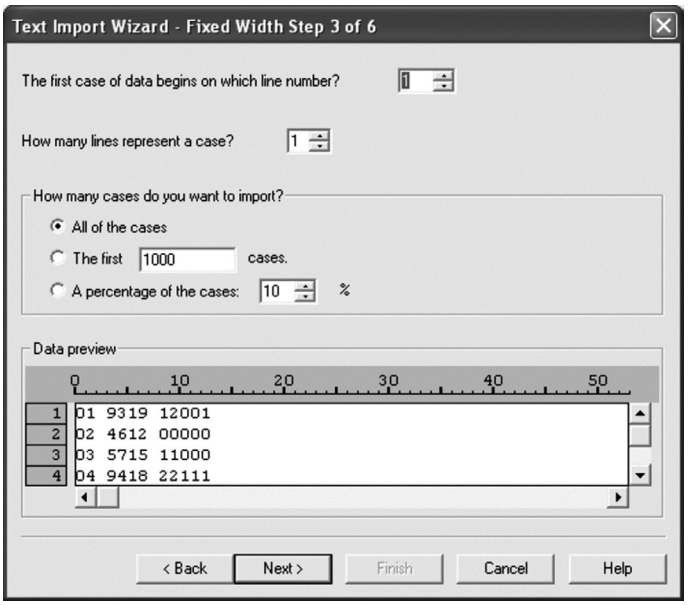

Figure 5.4 Text Import Wizard Third Dialog Box

Click the **Next** button to go to the fourth step. Click in the data view area to insert vertical lines indicating where each variable starts. You may need to refer to the codebook to remind yourself of the column positions for each variable. The dialog box will look like Figure 5.5.

Click on the **Next** button to go to the fifth step. Then click on each variable in the data preview, and then enter the variable name in the dialog box. Figure 5.6 shows the dialog box after the first three variables have been named. Continue the naming process until all the variables have been named.

Click the **Next** button to go to the final step. If you wish, you may save the file format for future use (for example, if you were going to repeat the same survey in the same format on more than one occasion), and you may paste the syntax to the Syntax Editor if you wish. The dialog box will look like Figure 5.7.

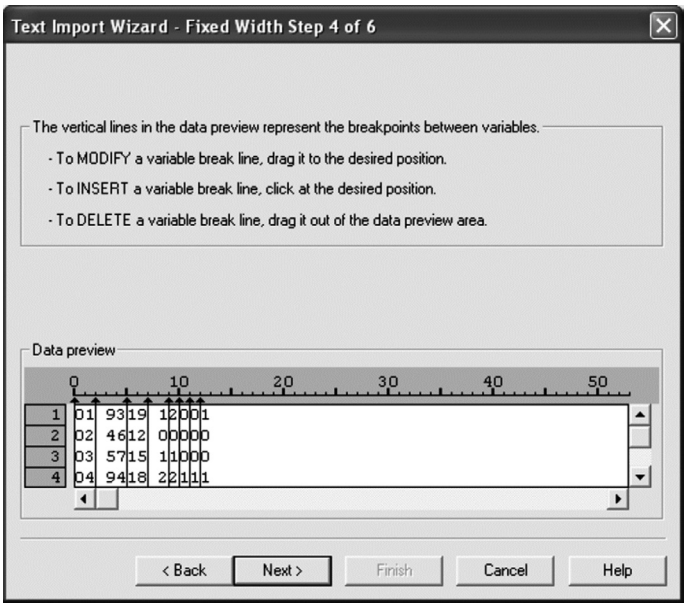

Figure 5.5 Text Import Wizard Fourth Dialog Box

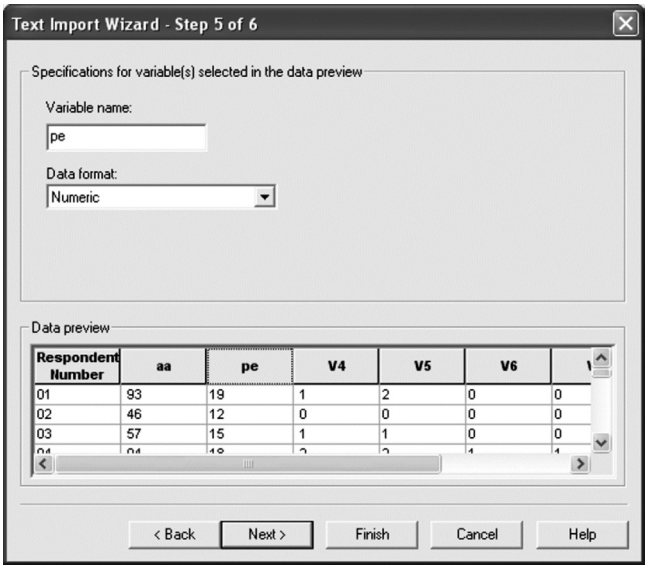

Figure 5.6 Text Import Wizard Fifth Dialog Box With First Three Variables Named

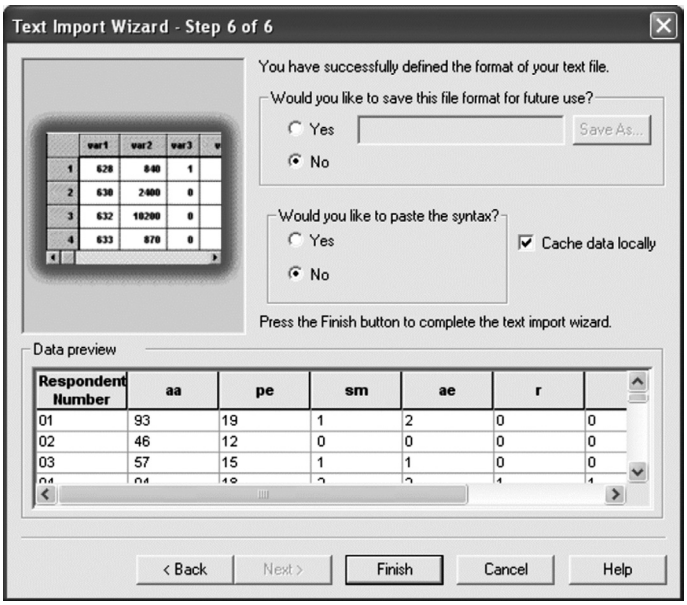

Figure 5.7 Text Import Wizard Sixth Dialog Box

Click the **Finish** button to read the data. Take a look at the Data Editor to confirm that SPSS has read the data from the text file and that they match the first five records of the Wintergreen dataset. At this point, you can save the file, which will be written as an SPSS-format data file.

IMPORTING FILES FROM OTHER SOFTWARE PACKAGES

SPSS is capable of recognizing data that have been entered and saved using other software packages. For example, imagine that the Wintergreen data have been entered using Microsoft® Excel. From the **File** pull-down menu, select **Open** and then select **Data . . .** You will see a dialog box allowing you to browse for the file you wish to open. At the bottom of the dialog box is a selection field called "**Files of type:**". From this list you can choose **Excel** *(*.xls)*. SPSS will then recognize that it is reading an Excel file. As SPSS is opening the Excel file, it will present you with the **Opening File Options** dialog box. Note that if the first row of the Excel spreadsheet contains the names of the variables

rather than data, you will need to select **Read variable names** in this dialog box. Also note that SPSS reads only one sheet at a time from the Excel workbook. If you have multiple sheets in a workbook that are related to one another, SPSS can read them using the Database Wizard (accessed from the **File, Open Database, New Query . . .** pull-down menus). It is also worth noting that many database and spreadsheet software packages can save (or export) data as an ASCII file, and you already know from the previous section how to read such a file.

MERGING DATA FILES: ADDING CASES

Sometimes data for a study are collected at different times, and sometimes they are entered at different times or by different people. In either of these cases, you may need to combine data files. The first method of combining files we will discuss involves adding cases to a file (also known as *appending* files). Imagine that data have been collected for an additional 50 students for the Wintergreen study. These data have been entered and saved as an SPSS data file. We would now like to combine this new file with the original Wintergreen data file, so that we end up with a single data file with 100 cases. To do so, first open the first data file. Next, from the **Data** pull-down window, select **Merge Files**, and then select **Add Cases . . .** The dialog box shown in Figure 5.8 will appear.

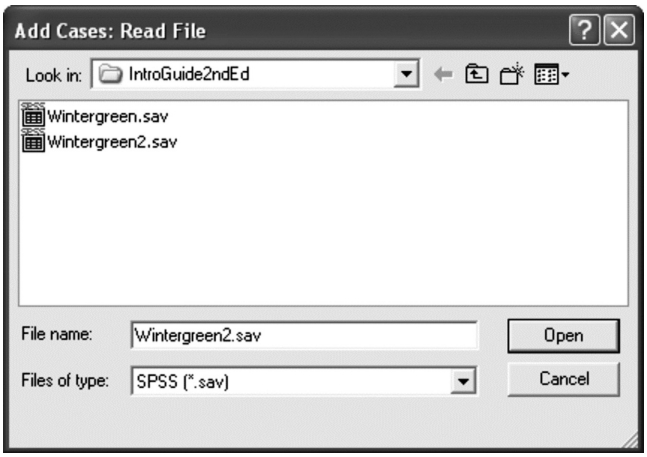

Figure 5.8 Add Cases Dialog Box

Enter the name of the file (in this example, the file name is "Wintergreen2.sav") with the cases to be added, and then click the **Open** button. The dialog box shown in Figure 5.9 will appear.

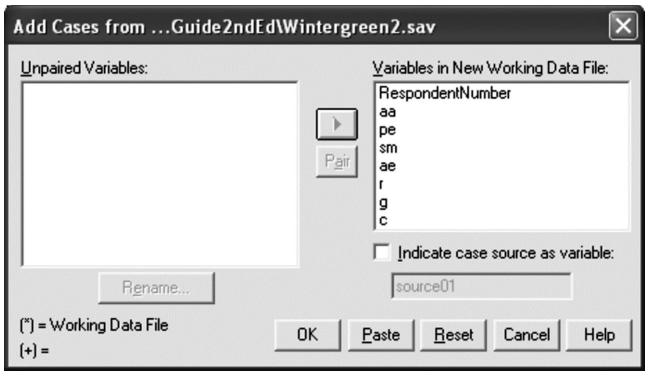

Figure 5.9 Add Cases Dialog Box

Click the **OK** button to add the cases in the second dataset to those in the first. The Data Editor window will now contain the total number of records. You may want to save this new combined file (perhaps giving it a new file name) in order to have a single data file with all the records (and thus being able to skip the step of appending one file to another the next time you want to conduct an analysis using all of the cases). It is also a good idea to list a few cases using the **Case Summaries . . .** command to see that you have correctly instructed SPSS how to merge the data.

MERGING DATA FILES: ADDING VARIABLES

There is another way of combining two data files that occurs when data on additional variables have been collected for the same persons who are already in the study. Imagine, for example, that a new variable is measured for the same 50 students in the Wintergreen study. For example, the new variable could be high school grade point average, and it could be stored in a file called "Wintergreen3.sav" (the variable Respondent Number would also be included in this file). What we would like to do is create a single data file with 50 cases and all of the variables.

First, the data in each file must be saved as an SPSS-format data file. Each file must also be sorted in ascending order by some *key* variable (in this case, the variable RespondentNumber). To sort a file, from the

Data pull-down menu select **Sort Cases . . .** The dialog box shown in Figure 5.10 will appear.

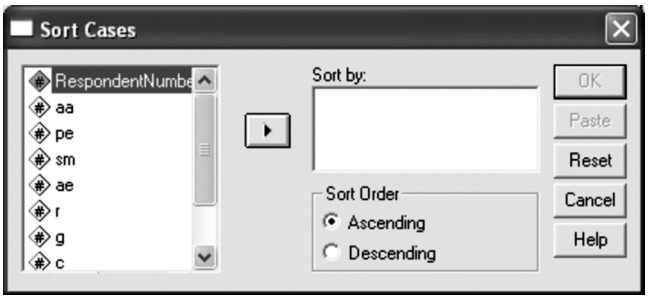

Figure 5.10 Sort Cases Dialog Box

Click on the variable in the list on the left that you want to use as the key (in this case the variable "RespondentNumber"), and then click on the button with the right arrow. Then click the **OK** button, and finally save the data file. Do this for both files, using the same key each time. Sorting the data and then merging them using a unique key ensures that the data from the two files are merged so that the data for the first respondent that resides in the first file is matched with the data for the first respondent that resides in the second file. The data are also similarly matched for the second respondent, the third respondent, and so on. After all, analyses would produce incorrect results if the data from the first respondent that is in the first file were matched with data from the second file that belonged to some other respondent.

Once the two data files have been prepared, open the first file. Then, from the **Data** pull-down menu select **Merge Files**, and then select **Add Variables . . .** You will see a dialog box that asks you to enter the name of the second file to be merged. Enter the name of this file and click the **Open** button. The dialog box shown in Figure 5.11 will appear.

Next, select **Match cases on key variables in sorted files**, click on the key variable (in our example, this would be the variable "RespondentNumber") in the box on the left, and then click on the button with the right-hand arrow that is just to the left of the **Key Variables** box. Click the **OK** button to merge the two files. As before, you may want to save this new combined file (perhaps giving it a new

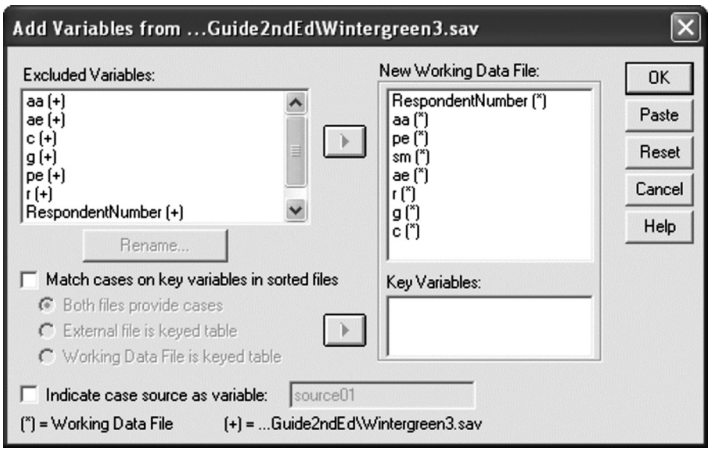

Figure 5.11 Add Variables Dialog Box

file name) to have a single data file with all the records (and thus being able to skip the step of merging the two files the next time you want to conduct an analysis using all of the cases). Again, it is also a good idea to list a few cases using the **Case Summaries . . .** command to see that you have correctly instructed SPSS how to merge the data.

EXERCISE FOUR

Iversen and Norpoth (1987) presented hypothetical data gathered in an effort to answer the question, "Does the mass media raise the public's concern with the economy by their coverage of economic news?" The data were gathered in an experiment in which subjects were randomly assigned to an experimental and a control group. Each group watched a television newscast made up of actual stories shown on the evening news. The experimental group watched a newscast that included a story on the state of the economy, and the control group watched a newscast that did not include this story. Members of each group then filled out a questionnaire that included a 10-point rating scale used to measure the importance subjects placed on the "state of the economy."

(Continued)

Suppose that the data have been entered into two different files by two different people and that the first person entered the data for the experimental group and the second person entered the data for the control group. To conduct the data analyses, the two datasets need to be integrated into one.

Using the data below, create and save a dataset for each of the two groups. Then append them into one dataset. List the data to confirm that the data were merged correctly.

Control Group Subject		Experimental Group Subject	
Number	Rating	Number	Rating
01	5	06	7
02	4	07	5
03	4	08	6
04	4	09	6
05	3	10	6

EXERCISE FIVE

In Exercise Three, we computed an average based on three test scores. Suppose that a fourth test is given and stored in a separate dataset. Using the following data, create this new dataset, and then merge the two datasets into one that contains all four test scores for each student. List the data to confirm that the data were merged correctly.

Student	
Number	Test 4
01	80
02	75
03	95
04	80
05	85