Chapter 2

Software Options for Entering Data

Before talking about the different types of software you can use for entering data, it's important to understand a few things about the basic structure of data sets. Statistical data can be conceptualized as a matrix, a grid of rows and columns, in which each row usually represents a single case in the sample, and each column reflects a single variable. As mentioned before, *cases* are the entities that are observed in the study, whether people, members of other species, industrial products, websites, or whatever. *Variables* are the phenomena on which cases are observed and categorized or quantified.

There are different types of variables possible, the most common of which are numeric, text, and date-time variables. Some data entry systems allow you to classify each variable before data entry occurs. This feature offers some protection against entering incorrect information. For example, if a variable has been defined as numeric, attempting to enter a nonnumeric value will be flagged as an error.

Text variables—also called string, character, or alphanumeric variables—allow values composed of letters, punctuation symbols such as exclamation points, numbers, or any combination of these characters. However, when a number is entered for a text variable it is treated simply as a label, not as a true number. For example, social security numbers and zip codes are really labels that just happen to be represented using digits rather than letters and should be treated as text variables.

Date-time variables are, as one might expect, used to enter time and/or date information. Note that when I refer to a time variable, I am referring to the time of day (e.g., 6:15 AM), not elapsed time (e.g., 34 seconds). Elapsed time is stored, instead, as a numeric variable; you must remember whether it is measured in seconds, minutes, or some other unit of time.

Some programs distinguish between date and time variables; others treat them as the same variable type. Date-time variables can accommodate different display formats (e.g., "3/11/62" versus "Mar 11, 1962") and can be used to compute time intervals accurately. For example, the interval between the date variables BirthDate and DateOfTesting can be used to compute the variable Age at the time that data entry took place. The software is able to perform computations using dates or times or recognize different formats because date and time variables are actually a special type of numeric variable. For example, Excel stores the date "12/11/2003" internally as 37966, the number of days since the end of the 19th century.

Tip: Before you get started with data sets, identify each of your variables as numeric, text, or date-time, which are the three most common types of variables.

I'll get to the topic of why I consider Excel in the Microsoft Office suite, or at least a similar spreadsheet program, to be the best single option for data entry shortly. First, I'm going to introduce some alternatives that may be useful in certain situations or that may be preferred by some users. These include entering data directly into the statistical software, using database software to create data files, and storing data in text files.

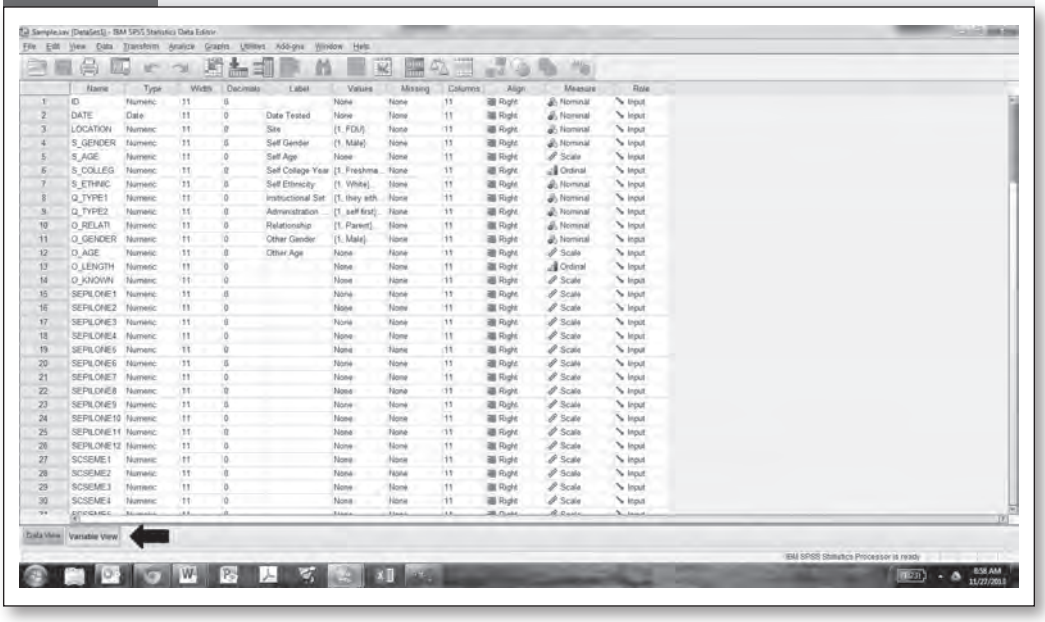
STATISTICAL SOFTWARE

All of the major comprehensive commercial statistical programs include a facility for creating data sets. For example, Figures 2.1 and 2.2 are from the SPSS Data Editor. Figure 2.1 is the Variable View window, Figure 2.2 the Data View window. You can switch between the two windows using the tabs at the bottom of the Data Editor, indicated by the arrow at the bottom of Figure 2.1. The Variable View window provides information about the variables contained in the data set while the Data View window presents the data set itself.

The Data View window is pretty typical of how statistical programs organize data. The data are arranged as a grid or matrix in which columns represent variables and rows represent cases. Variable names (ID, Date, etc.) appear above the first row of the grid, case numbers to the left; that is, variable names and case numbers are treated as distinct from the matrix of variable values. Each cell in the matrix contains the value of some variable for some case.

The Variable View window provides a list of the variable names and basic information about each. For example, the Type column indicates whether the

Figure 2.1 The SPSS Data Editor Variable View Window

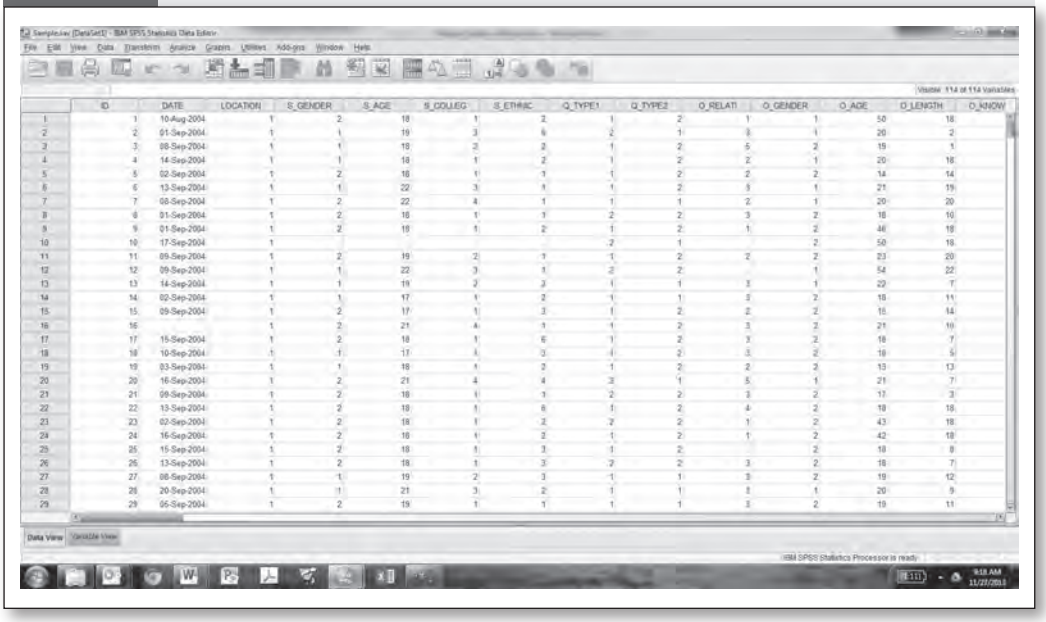


variable has been defined as a numeric variable, text variable (called *string variables* by SPSS), or date-time variable. It's important to type variables accurately. One variable may consist of nothing but numbers, but if it has been incorrectly identified as a text variable, then the statistical software will not allow you to conduct analyses that require numeric data. Another variable incorrectly identified as numeric will enter a missing value for any text entries. The other columns in the Variable View tend to be less important, so I won't discuss them here. A couple of them will deserve mention in Chapter 7 when I discuss transferring data from Excel to SPSS.

It might seem logical to use statistical software to create the data sets that you will later analyze statistically, and many students assume it's the best option. In fact, for small data sets, using SAS or SPSS to enter the data may be reasonable. After all, once the data are entered they're in a format understood by the software, so there are none of the problems I will discuss later that are associated with moving information from one program to another.

For larger data sets, however, statistical software is rarely the best choice. Developers of statistical software are usually more interested in data analysis than data entry, and the options for entering the data directly into the statistical

Figure 2.2 The SPSS Data Editor Data View Window



software aren't very sophisticated. To cite just one example, suppose you want to create a sequence of 30 variables representing the 30 items of the McGrath Make-Believe Questionnaire. You might want to call these variables MMBQ1, MMBQ2, and so on up to MMBQ30. Creating these 30 variables in SPSS would require entering each variable name individually. As you will see, Excel will allow you to create all 30 variable names in a matter of seconds.

One final disadvantage to using the statistical software for data set creation is limited availability. In any study, it's usually the case that only one or two people are involved in analyzing the data, but the data entry process can be spread among many people, especially if the data set is a large one. Entering the data directly into SPSS or SAS means access to the software can become a limiting condition to the number of people involved in the data entry. In my institution, every university computer offers several different statistical software programs, but this is not always the case. Even if it is, students often want to enter data at home or at some other location (assuming, of course, that any materials capable of identifying the members of the sample, such as the consent form, are removed first). Using the statistical software for this purpose will require a license that allows installation on the student's computer—an option many universities do not pursue because of the expense—or purchasing a student version of the software.

Tip: Statistical software is probably only a good choice for data entry if you're creating a fairly small data set.

ACCESS AND OTHER DATABASES

A more widely available alternative to statistical software is a database, such as Microsoft Access or OpenOffice Base. Databases have some desirable features as a data entry option. They create data files that are similar in structure to those used in statistical analyses, though they tend to use different terminology. In Access, for example, the data set is stored as a *table* within a database file, which can contain multiple tables. The term *field* is often used to refer to variables, and the rows are usually called *records* rather than cases.

Despite differences in the wording, the similar structure means that database tables can be imported into statistical software with a relatively high degree of fidelity. Figure 2.3 provides the same data set shown in Figure 2.2 in an Access data table.

Figure 2.3 An Access Table

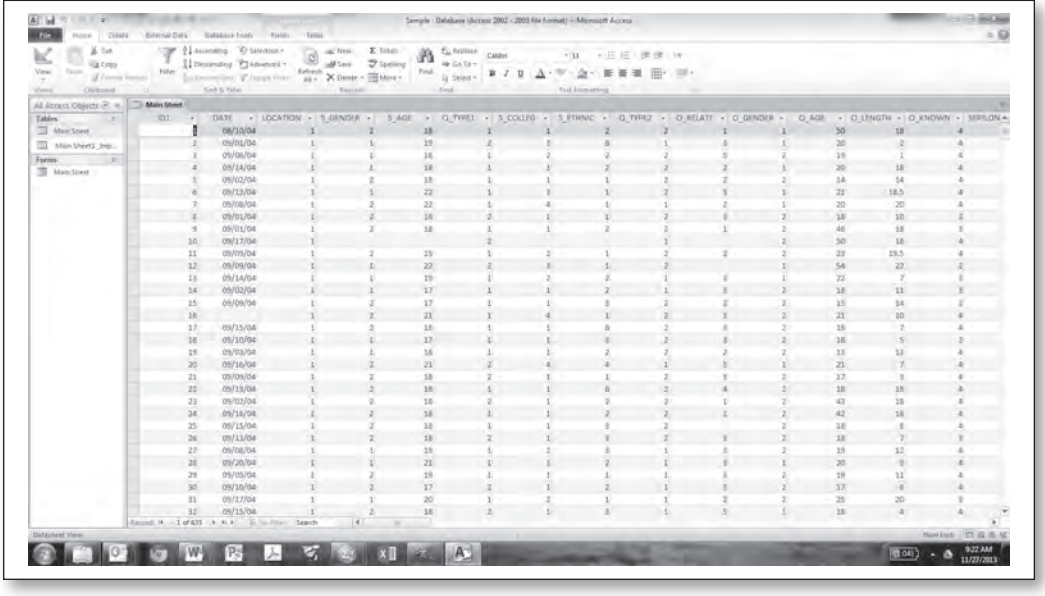
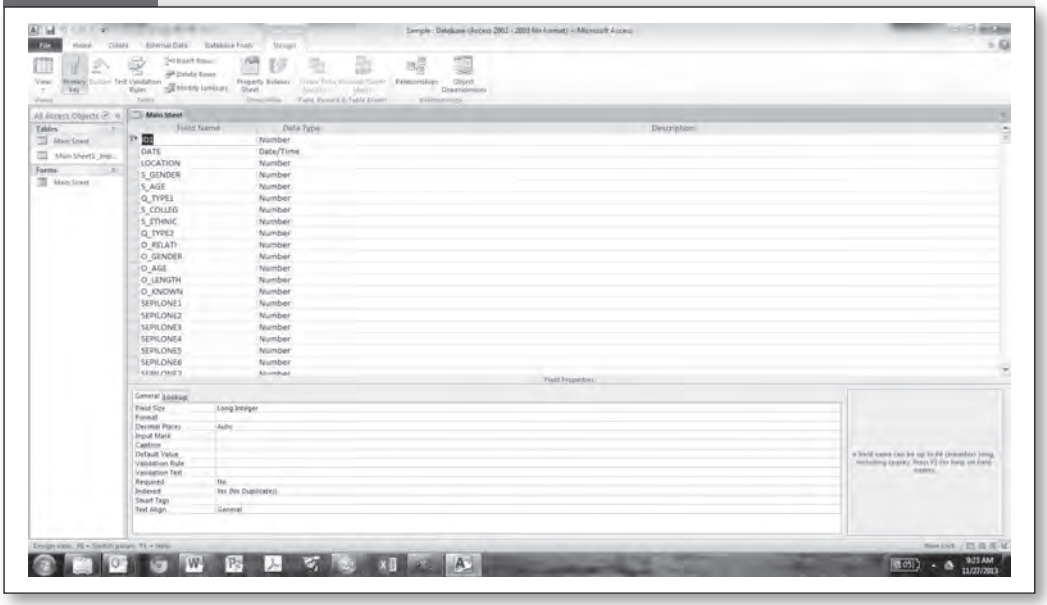


Figure 2.4 Access Table Design View

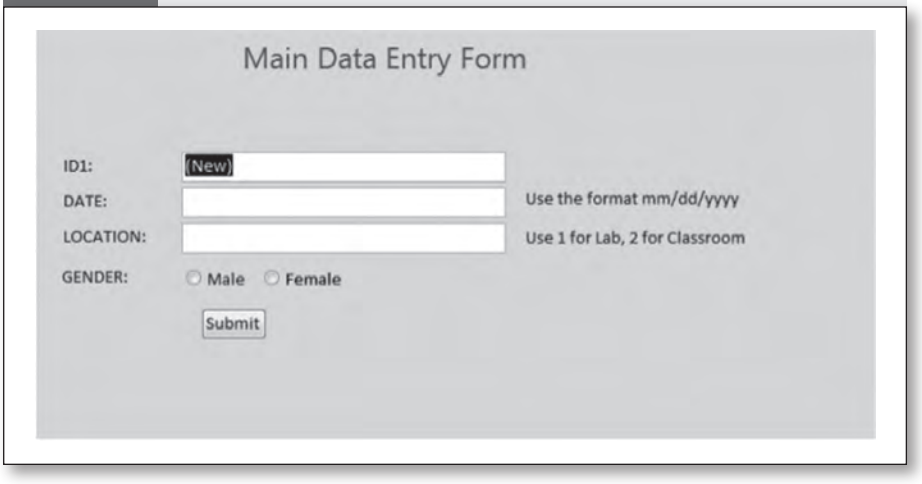


Database software also allows you to identify the type of information that can be entered into any field as numeric, date-time, and so on. Figure 2.4 shows the Design View of the table in Figure 2.3. There are substantially more options available for defining fields in Access than is true of variables in SPSS.

Databases offer several other very neat options. They are particularly flexible if your goal is to generate snazzy looking reports summarizing the data, though this functionality is usually of more interest in business than it is in scientific research. A potentially more useful tool is the ability to create a *form*. A form represents a "front end," or data entry screen, for the database. When you go to a store and the clerk enters your name, address, and other information into predefined fields, he or she is usually interacting with a database through a form designed for that purpose.

Forms can be particularly helpful when data have to be entered by people with very little training in research, such as employees of community agencies. The front end can include instructions, branching based on prior information (e.g., a field for current grade only becomes available if the data entry person has previously indicated the participant is still in school), and other helpful features that reduce the number of errors when data are entered. Figure 2.5 is a very simple data entry form for several of the fields in Figure 2.3. Forms are

Figure 2.5 A Sample Access Data Entry Form



made up of *controls* such as text labels, radio buttons, checkboxes, and so on. Controls limit the type of information that can be entered. Pretty much every characteristic of these controls—fonts, colors, appearance—can be manipulated, and a wide variety of controls is available in addition to what is presented in this example (e.g., hyperlinks and dropdown boxes). Forms are incredibly flexible as a method of data entry.

However, before you decide to create a front end to simplify matters for the people entering your data, be aware that the process can be time consuming, especially if you want to get it just right. Of course, the more data to be collected, and the less trained the people entering the data, the better the justification for taking the time to make a well-designed front end.

Databases are more generally available than statistical software. In particular, databases are used extensively on the Internet for data collection and storage. The major statistical programs also include facilities for importing data from at least some database programs. However, as a method for data entry, databases suffer from some of the same inefficiencies as statistical software. For example, field names usually have to be entered individually (there are ways to create sequential variable names, but you have to be an advanced user). Furthermore, in the personal computing world, databases are not as widely available or generally familiar to users as the next two options. For this reason, though website designers frequently rely on databases to collect data on web pages, it's often the case that the data are converted to a more familiar format, such as a spreadsheet file, for distribution to others. In universities, databases are not a popular option for entering research data.

Tip: A database is probably only a good choice for data entry if you want a sophisticated data entry form to reduce errors by minimally trained personnel.

TEXT FILES

Once upon a time, text files were probably the most common method for transferring data between people and programs. Text files play various important roles in statistical analysis, so it's worth taking the time to learn a little about them even if you don't plan to use them for creating data sets.

To help understand the concept of a text file, consider what's contained in a word processor file. When you create a document in a word processor such as Word or Pages, you create text—the letters, numbers, and punctuation that comprise the words, sentences, and paragraphs of the document. Those characters are coded inside the computer using a standard numeric code. For example, all computers code the character *capital F* using the same numeric code, which is different from the numeric code for *lowercase f*.

Word processor files also contain information *about* the text: "the margins for this page are one inch on each side," or "bold this text," or "center the page number at the bottom of each page." This information about the text, sometimes called *metadata*, can be quite extensive. If you open a blank document in Word and save it without entering even a single character, the resulting file contains about 13,000 bytes of information, most of it information about margins, fonts, and so on.

The metadata is coded uniquely by each word processor. The symbols used to set the margins in Word are not the same as those used in Pages. It is true you can often create a file in one word processor and open it in another, but that's because the software developers add a special code, called a *driver*, for translating files created in another word processor. If you try to open a file in Microsoft Word that was originally written with the word processor WordPerfect, it does not open immediately. Instead, Word first uses its WordPerfect driver to translate the WordPerfect metadata into Word metadata. You may sometimes find errors in the conversion because there can be inaccuracies in the driver.

This distinction between text and contextual information applies to statistical and database files as well. An SAS data file contains the text that makes up the names of the variables and each case's value for each variable. The file must also contain contextual information that associates each value with a particular variable and case. This information is used to create the grid that you see when you look at the data in the SPSS Data View window or the Access table. As in the case of Word, this contextual information is coded differently by different

programs. A data file created using SAS can only be read by SPSS if SPSS includes a driver for that type of SAS data file.

In contrast, a text file contains only characters with no metadata about those characters. That is, it contains letter, numbers, and other symbols found on the keyboard, but no information about how the text is to be formatted or organized. If you save an empty text file with no characters added to the file, this file will be listed as 0 bytes long: Without text it has no content. Text files go by various names besides those already mentioned, including plain text files and DOS text files. The first generation of numeric codes used in desktop computers to represent the characters was called the American Standard Code for Information Interchange, or ASCII, which was later replaced by a more extensive set of characters called Unicode. For that reason, text files are also sometimes referred to as ASCII text or Unicode text files. I'll introduce still other terms used to refer to text files later as they become relevant.[1]

Here is a sample of what a text file used to store data might look like:

```
ID, Gender, Age, Q1, Q2, Q3
1, "M", 23, 1, 3, 4
12, "F", 18, 2, 3, 3
```

That is, the first line contains the variable names, and each subsequent line contains data for a single case. A character—called a *delimiter*—is used to separate the values. In this case, the delimiter is a comma and a space. Because this is a text file, there is no information in the file that says, "For the second case, the value for variable *Gender* is *F*." However, most statistical software will be able to use the position of values in the text file to associate the value *F* with the variable *Gender* and case *2*. This is one way SPSS or SAS can import data stored in a text file.

The most common delimiters are the comma, the space character, and the tab character (yes, space and tab are characters in the ASCII code). If you find yourself, for whatever reason, working with text files, I strongly recommend you use tabs, not commas or spaces, as delimiters. For one thing, when you open the file in Word, the variables all line up in columns, though sometimes you have to adjust things a little to make that work. Also, it is possible for

1. Just to be precise, I'll mention there are exceptions to this distinction. HTML, the language used for web pages, is transmitted over the Internet as text files, in that the contents are restricted to Unicode characters, but these files include both the words that appear on the web page and special code for formatting information that your browser interprets to figure out what the page should look like. This special code represents text-based metadata. The same is true for wikis. This is why web pages can be transmitted so efficiently over the Internet.

values of text variables to include commas and/or spaces (e.g., when the variable is an address), and tabs make the boundaries between variable values clearer. This is also why it is usually recommended that in text files, you enclose values of text variables with quotation marks.

There is a second method for storing data in text files. This method might format the data above as follows:

```
01M 23134
12F018233
```

That is, the first two columns are always devoted to the ID number, the third column is Gender, columns four to six are for Age, and so forth. This is often called *fixed field* or *fixed width formatting*, versus the free-field format I described above. Notice that for the text variable Gender in the delimited file, values were enclosed in quotation marks. I mentioned this is recommended to prevent confusion if values of text variables include commas, spaces, or whatever the delimiter is. The quotes weren't necessary for the fixed field data because the software will know exactly which columns contain which variables. Notice that a multicolumn variable can be filled with zeroes—as I did for the ID value 1 and the Age value 18—or with spaces, as I did for the Age value 23. Either option is acceptable.

To read fixed-field files, the statistical software has to identify which variables are in which columns. Though pretty much every major statistical program offers a method to read fixed-field text files, the fixed-field format has dropped out of fashion. I still see it used from time to time because it results in smaller files than delimited data.

You can use any word processing software to create a text file. For example, when you click on *Save As* in Microsoft Word, you will find a dropdown box called *Save as Type*, and one of the options is Plain Text. Plain text files usually have the file extension *.txt* by default, though some programs use *.dat* instead.[2]

---

2. In case you are unfamiliar with the term *file extension*, in most operating systems the name of a file actually consists of two parts: the base file name and a file extension. The file extension is a suffix, usually three or four letters, added to the base file name to provide information about the type of file. It is generally separated from the base file name by a period. For example, if I save a Word document with the base file name My.Data, Word adds the extension .docx, so the complete file name is My.Data.docx. The extension *docx* is what the operating system uses to identify this as a Word file. By default some operating systems hide the extension. I have never really understood why, as I find the extension sometimes helps me figure out what's going on when I have problems with a file. If your computer hides file extensions, I would recommend you look online to find out how to get your operating system to show them instead.

If the delimiter is a comma, another option for the extension would be *.csv*, for "comma separated values." Be forewarned, though, that word processors can cause problems with your text files. For example, Word's Autocorrect automatically converts straight quotes (") to curly quotes ("and") that will confuse your statistical software. You can turn off the automatic formatting. Alternatively, you may be better off using a text editor that automatically saves files as plain text. Windows includes the text editors Notepad and Wordpad, while TextEdit is available on Mac computers. Many more sophisticated editors are available online. Emacs is particularly highly recommended, and despite the name is available for pretty much every operating system.

In theory, any statistical software should be able to read data stored in a text file, and some programs (e.g., R and many of the structural equation modeling [SEM] programs) expect data as a text file. However, programs differ in how they expect the file to be structured. Can your program draw the variable names from the first line of the text file, or do you have to list the variable names in the commands used to read the data? Does your program assume the first line contains the variable names, or do you have to state that explicitly in the commands? Do the variable names need to be in quotation marks? Do string variable values need to be in quotation marks? What delimiters are permitted? The answers to these questions vary across statistical software, and to work with text files you will have to know what your program expects.

As a method of data entry, text files have an advantage over the prior options in that any computer with a word processor or text editor has at least one program that can be used to create text files. It's usually not the best option, though. It's easy to make errors when the variables are not organized neatly into columns, and automatic formatting in word processors complicates matters. For that reason, the potential for error is greater with text files than for any of the other options discussed. Even when the statistical software I'm using works best with text data, I usually create the data set using Excel because of its superior data entry and verification options; then I export the data set to a text file only when the data set is in its final form.

Before finishing this section, I want to bring up one other use of text files in connection with statistical software. I mentioned earlier that experienced statisticians tend to conduct their analyses using commands rather than dialog boxes. The major statistical programs all allow you to create files of commands that can then be run all at once (the standard techie lingo for this is "running commands in batch mode"). Such files are called *programs* in SAS and usually have the file extension *.sas*. They are also called programs in R and have the file extension *.R*. In SPSS they are called *syntax files* and are given the file extension *.sps*. In Mplus they are *input files*, with the file extension *.inp*. In all four cases, these files

are saved as text files. As your skills as a statistician improve, you will find command files to be a valuable tool for making your work more efficient.

For example, suppose you want to run 30 separate multiple regression analyses with minor changes from one to the next. You could repeat the analysis 30 times using menus and dialog boxes, but that would quickly become tedious. There is a much more efficient option. You can use the dialog box the first time to design the analysis. An interesting feature of statistical software is that when you use a dialog box and click on *OK* or *Submit,* the first thing the software does is generate the commands that correspond to the analysis you set up. Depending on the software, there are various ways to access those commands. You can then copy the commands to a word processor or text editor 30 times (though if you use a word processor, keep in mind the warning I raised earlier about automatic formatting of some characters), edit them to reflect the changes from one analysis to the next, and run all 30 analyses in a batch. The commercial statistical programs even incorporate facilities for this copying and editing within the software, though, as you can imagine, they are not as flexible as a full-featured word processor.

There are several advantages to running your analyses in this way:

1. It is simply much faster than using the dialog box repetitively.
2. It creates a permanent record of your analyses. My students know to bring me their command file if they are having problems with an analysis. Even when the analysis runs perfectly, I often find it helpful to reevaluate the analysis, and having the record of what we've done in front of me facilitates the process.
3. If you later find you need to change some analyses, you can edit the command file and repeat the analyses much more efficiently than is possible with the dialog boxes.

The moral of this story is that you may want to use the more intuitive dialog boxes to help you see what the commands look like, but any complex analysis is better conducted using a command file rather than dialog boxes, and using command files requires dealing with text files.

Tip: Text files are not recommended as a way to store data or create data sets. However, I still recommend that you read this section because text files play several important roles in most statistical programs. Also, if you ever do receive data in text format, you'll need the information in this section to understand the later section on how to import text data into Excel, and you can then use Excel tools to look for problems in the data.

EXCEL AND OTHER SPREADSHEETS

In the end, I recommend a full-featured spreadsheet program such as Microsoft Excel, iWork Numbers, or OpenOffice Calc over all the alternatives. At one time, this recommendation wouldn't have made much sense. Spreadsheets involve placing information into a grid very similar to the data matrix in Access and SPSS. However, spreadsheet software was originally developed to simulate the grid found in accounting spreadsheets. These don't assume any particular organization for the contents of the grid, so spreadsheets, by default, do not make any assumptions about whether values in the same column or same row are associated with each other.

Even so, there are several reasons why software originally meant for tracking financial accounts now represents the best general choice for statistical data entry:

1. Of all the options listed so far (other than word processors or text editors), the spreadsheet is the most widely available type of software. The Microsoft Office suite alone is installed on over 300 million personal computers. Software is available for editing spreadsheets on tablet computers, iPads, and even on smart phones (I have done this, and it works in a pinch). Cloud computing expands the options even further, making it possible to store, and in some cases edit, the spreadsheet on the Internet. One cloud option is a storage service such as Dropbox or iCloud, which allows each data entry person to edit the file in the Excel software on his or her own computer but then store the results on the Internet. There is also growing interest in cloud application services such as Google Drive or Office 365 that allow editing of the data using online spreadsheet software. Either way, changes made by one data entry person immediately become available to all. I find it amazing when a group of people can sit in a room together, each connected to the Internet, collaboratively designing a data entry spreadsheet using Google Drive, with each person's changes appearing on everyone's computers simultaneously, then maintaining the document online so each researcher can add data whenever it's convenient. The ubiquity and familiarity of spreadsheets is what makes this kind of collaboration possible.

2. Because of that ubiquity, familiarity with Excel data entry is a desirable characteristic in potential employees in many fields, something that is not often true about other methods of data entry. Readers who do not

envision a future as a psychological researcher might as well enhance a marketable skill through data entry experience.

3. Many students never work on more than one or two research projects, so learning new software just for the purpose of data entry represents an inefficient use of their time. Most computer users have had some experience with spreadsheets, so even research novices can be entering data quickly.
4. Excel is used enough for data entry that every major commercial statistical program includes drivers for directly importing Excel spreadsheets. Similarly, Excel can be used to import data from various sources, including databases and text files. Data contained in a table on a web page can simply be copied and pasted into an Excel spreadsheet, and data collected online in a database is often converted to a spreadsheet for purposes of subsequent analysis.
5. Data online is often made available for download in a spreadsheet, such as that provided at http://www.census.gov. When I'm looking for examples of data sets online, I will put in *xls* or *xlsx* (the most common file extensions for Excel spreadsheets) as one of my search terms and frequently find data all set for analysis.
6. It was noted earlier that Excel also offers options for computing descriptive and some inferential statistics. In a pinch, Excel can be used to generate basic statistics. I sometimes find these statistical tools helpful to compute a quick correlation or set of means.
7. Developers of spreadsheet software recognized their programs were being used extensively for data entry, and they have added functionality intended to enhance data entry and manipulation. Excel provides a remarkable array of tools that can ease the task of data entry, many of which have been added specifically for this purpose. In fact, one of the top-level menus in recent editions of Excel is titled Data.

In the chapters that follow, you will learn about this remarkable set of tools intended to help improve your data entry and verification. I suspect by the time you're done you will agree it's a remarkably effective and efficient way to build good data sets.

CHECK YOUR UNDERSTANDING

C2-1. Give an example of a research variable that would best be represented numerically, one that would best be represented by text, and one that would best be represented by a date variable.

C2-2. Current marital status is a variable usually conceptualized as having five values (Single, Married, Separated, Divorced, Widowed). Would you represent this variable with a text or numeric value? Explain your choice.

C2-3. If I have two dates in an Excel file, and I subtract one date from the other, how could I interpret the resulting difference? What does it mean if the difference is positive? What does it mean if the difference is negative?

C2-4. List at least one pro and one con for using each of the following to create data sets. I'll warn you, this may require thinking beyond what was said in the text:

1. Statistical software
2. Database software
3. Text files
4. Excel

C2-5. List three uses of text files in the context of research and sharing information across programs.