At one level ‘data’ are the world that we want to explain ... at the other level, they are the source of all our troubles.


The ethnic theory of plane (and data) crashes

In his compelling book Outliers: The Story of Success, Malcolm Gladwell presents what he calls the ethnic theory of plane crashes: ‘[P]lane crashes rarely happen in real life the same way they happen in the movies. Some engine part does not explode in a fiery bang. The rudder doesn’t suddenly snap under the force of takeoff... Plane crashes are much more likely to be the result of an accumulation of minor difficulties and seemingly trivial malfunctions’ (Gladwell, 2008, p183).

This is true for data quality problems. Rarely does the ‘computer’ corrupt data, although a computer virus may do so or even delete data. Instead, small mistakes at various stages can seriously compromise data quality. The research question, sample selection, questionnaire design, enumerator training and survey implementation, all of which have been discussed in preceding chapters, impact data quality. However, there are computer specific sources of errors that impact data quality, the most critical being data entry and storage. A poor data entry system is a gateway to an unnecessarily large number of data entry errors, while failing to organize and archive data in a way that they can readily be retrieved and analysed can render them unusable.

The purpose of this chapter is to look at these computer-related data problems. Drawing on the Center for International Forestry Research’s (CIFOR) experience with the Poverty Environment Network (PEN), this chapter identifies potential sources of errors and makes some suggestions on
ways in which these errors can be prevented or minimized. Recognizing that some errors may still slip through and that some accurately transcribed values may have been recorded in error, approaches to data validation and checking are also discussed.

Because, as will be argued later, working with software one is familiar with tends to minimize errors, the ideas presented in this chapter will largely be software independent. However, for some illustrations, Stata (StataCorp, 2009) – the program used for PEN data management – will be used. Also, the following assumptions are made: (a) that data will be primary data collected using a questionnaire, such as a household survey; (b) that these data are mostly numerical, a mix of quantitative and qualitative information with the later being assigned numbers or codes for data entry; and lastly (c) it is assumed that text-heavy sections of the survey such as notes made during or at the end of survey will simply be compiled into some sort of narrative and are not checked for quality.

The chapter proceeds as follows. First it gives a brief definition and examination of the concepts of data and data quality as they will be used in the chapter. Second, it examines the building blocks critical for ensuring data quality – namely coding, data entry and data validation/checking. Third, the chapter ends with a summary and, through the conclusion, synthesizes the discussion into a few take-home messages.

**Improving data quality**

Data and information are often used interchangeably yet they are distinct (Audit Commission, 2007). In this chapter, *data* will refer to numbers and words that are yet to be organized or analysed to answer a specific question. These are sometimes referred to as raw data, in other words, the numbers and text recorded on the questionnaire and later transcribed to the computer.

Data quality has many dimensions, however, most of these can be distilled into the simple yet comprehensive definition: ‘Data are of high quality if they are fit for their intended uses in operations, decision making, and planning. Data are fit for use if they are free from defects and possess desired features’ (Redman, 2001, p73). This fitness for use comprises two characteristics; possession of desired features and freedom from defects.

The desired features include relevance, comprehensiveness, proper level of detail, ease of reading and ease of interpretation, while data are free from defects if they are accessible, accurate, timely, complete and consistent (Redman, 2001). These features have been, implicitly or explicitly, addressed in other chapters of the book. For example, timeliness of research is covered in Chapter 3, while
relevance, comprehensiveness and the proper detail are covered in both Chapters 3 and 4. Ease of reading and interpretation, which arguably has more to do with information (processed data), is addressed in Chapter 14.

Our focus is on data defects in terms of accuracy, completeness and consistency. A comprehensive treatment of the each of these concepts can be found in, among others, Lee et al (2006) and Refaat (2007). The section below is largely drawn from these two references.

**Accuracy:** This is perhaps the most likely aspect of data quality that comes to mind when you ask scientists about data quality, and is often used to mean that data are free from errors. Refaat (2007) defines errors as an impossible or unacceptable value in a data field. Errors can be typographical (‘mlae’ instead of ‘male’), mathematical (income not equal to gross value less purchased inputs) or related to measurement (such as the location, area, quantities).

**Completeness:** Data completeness refers the degree to which data are missing. According to Lee et al (2006), it can be viewed from at least three perspectives: schema completeness, column completeness, and population completeness. Schema completeness is the degree to which entities are not missing from the schema. Attrition is a good example, where, for example, households that were originally part of the survey drop out in later rounds. Column completeness refers to the degree to which values are missing in a column of a table. This is the most common type of completeness problem in socio-economic surveys and the subsequent discussion on missing data in this chapter will mainly be about column completeness. Lastly, population completeness refers to the degree to which members of the population that should be present are not. This is a usually a sampling problem.

**Consistency:** Data consistency may be defined in terms of consistent values among different fields, as well as values of the same field over time (Refaat, 2007). For example, children should not be older than their biological parents. The area under agricultural production cannot be greater than the total area owned or operated by a household. It is common to have questions whose responses depend on earlier responses. For example, the response to the question on how severe the crisis was depends on the response to the question of whether or not the respondent faced the crisis. Questions like these are added to improved data quality and are good candidates for consistency checks.

### Data entry and quality checking: Work flow

Picture the start of fieldwork. The study site has been selected, the sampling frame determined and the sample drawn. Enumerators have been trained and the questionnaire has been pretested, improved and is now ready for use. A data
entry system is in place as is a basic codebook. The survey kicks off and questionnaires begin to come in. What next?

Trying to work efficiently and, given that it is technically feasible (you can use a computer), it may be very tempting to hit the proverbial birds with one stone: start entering the data and, in the process, scan the questionnaire for errors while also assigning codes for categorical data. This may create the illusion of saving time; however, doing this is not only inefficient – it is a recipe for errors.

It is inefficient because of interruptions that will, more often than not, arise because of illegible answers, gaps, and other inconsistencies, which will require that the questionnaire be returned to the enumerator for correction or even to revisit the household. The challenge is how to deal with the interruption. Do you pause data entry for that questionnaire until the errors have been addressed, or, do you discard what you have entered and start all over again? Pause and resume and you may enter the same data more than once, discard and you will have wasted precious time.

A more efficient and less error-prone way to work is to think of data entry and checking as a process consisting of four tasks: visual inspection of the survey instrument, coding, data entry and data checking. These tasks are best performed sequentially in a way that is akin to the flow chart of Figure 12.1.

**Figure 12.1** The process of data checking
**Visual inspection**

The goal of visual inspection is to make sure that all sections of the questionnaire that should be filled in have been filled in. Check that all standard information, that is, information that does not depend on the respondents (such as location, date of interview, household code and so on) is filled in.

Visual inspection is also the first step in ensuring that data are usable. It is not enough that the questionnaire has been filled in, the recorded responses should be clear and therefore make the questionnaire easy to transcribe to the computer.

Visual inspection is best done in the field because of the ease of making follow-up visits for clarification or collecting missing information if necessary. If you are working with many enumerators, then it is recommended that you have a debriefing session every evening where enumerators go through the questionnaires for these issues and share their experiences. This is a great learning opportunity, especially if one cannot be present during the interviews during the day.

**Coding**

Most data entry programs will accept both numbers and text; however, it is better to enter numbers. To illustrate, consider the hypothetical data set in Table 12.1 where, for one household, John Mayne, we have data on the main crop on his four plots. The variables are the name of the household head \( \text{name} \), plot number \( \text{plot} \), the sex of the plot manager \( \text{sex_pm} \), and the main crop grown on the plot \( \text{main_crop} \).

Looking at Table 12.1, one may immediately be drawn to the errors in the spelling of the name. While these have been exaggerated for illustration purposes, such mistakes are not uncommon especially when you have more than one person entering data. John Mane may stand out as an obvious mistake, however, as far as the computer is concerned, Mayne John, which reverses the name order is a different person as is john Mayne, which only differs because of the lower case in the name John.

<table>
<thead>
<tr>
<th>Name</th>
<th>quarter</th>
<th>sex_pm</th>
<th>main_crop</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Mayne</td>
<td>1</td>
<td>Male</td>
<td>Maize</td>
</tr>
<tr>
<td>john Manye</td>
<td>2</td>
<td>male</td>
<td>Cassava</td>
</tr>
<tr>
<td>Mayne John</td>
<td>3</td>
<td>Male</td>
<td>Bananas</td>
</tr>
<tr>
<td>John Mane</td>
<td>4</td>
<td>male</td>
<td>Potatoes</td>
</tr>
</tbody>
</table>
More interesting, however, is the variable ‘sex_pm’ where it appears that the only offending observation should be in the fourth row where there is an obvious gap between m and a, and yet, when we look at the data, we find that each entry of ‘male’ has been treated as a different category, see Table 12.2.

The reason all four have been treated as different is because of what is called ‘air’ or leading/trailing blanks. These are often non-perceptible spaces created when you hit the space bar before or after data. Regardless of where the air is, the effect is the same, it changes the entry. For a computer to consider two bits of text as the same, they have to be identical. This includes any leading or trailing blanks as well as the case of the letters, hence the treatment of male and Male as different.

These issues can be addressed: a variable can be trimmed to remove leading and trailing blanks cases of all observations can be standardized (all changed to lower). However, this creates additional work and is a potential source of errors.

As a preferred alternative, data can be entered as numbers.

### Advantage of using codes

Coding is the process of assigning numbers to text and the document containing all such codes as well as variable names and descriptions is known as a codebook or coding frame. Using codes instead of text has several advantages, which are summarized below:

**Error and ambiguity**: Numbers are less error prone than text. One can enter 10 instead of 1, but that pales in comparison to the errors one can make when entering text. There is also generally no ambiguity with numbers, 1 is 1 but, as shown above, small changes in text introduce ambiguity that impacts data quality. Are Cassava, cassava and cassava different crops? In the example above, if we had in addition to the household name assigned a numeric household code to each household, we would not have had to worry about spelling mistakes in John Mayne’s name. The same problem would be addressed if, instead of entering the sex variable as male/female, we used 0/1. Lastly, numbers are not case-sensitive, and tend not to have the ‘air’ problem common with text.

<table>
<thead>
<tr>
<th>Sex of plot manager</th>
<th>No.</th>
<th>Col %</th>
<th>Cum %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>1.0</td>
<td>25.0</td>
<td>25.0</td>
</tr>
<tr>
<td>male</td>
<td>1.0</td>
<td>25.0</td>
<td>50.0</td>
</tr>
<tr>
<td>Male</td>
<td>1.0</td>
<td>25.0</td>
<td>75.0</td>
</tr>
<tr>
<td>male</td>
<td>1.0</td>
<td>25.0</td>
<td>100.0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>4.0</td>
<td>100.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 12.2 Further illustration of the problem with using text rather than numbers when coding
Restrictions: Most data entry programs allow users to restrict entries in a given column to a range of values (for example, between 1 and 10) or a list (for example, 1, 2, 3). These restrictions, also known as validation rules, would prevent the entry of illegal values unless they are changed. It is harder to set validation rules with text even though you could for example restrict the length of the text entered or create a drop-down list with pre-entered text from which a user can select a response. While these can help, it remains much easier to work with numbers.

Ready to use: Data entered numerically are ready to use at subsequent stages. In the example above, entering male/female for the sex variable would require that the variable is converted to a numeric variable before using it in, say, a regression. Related to this point, it is advisable, when designing the questionnaire, to try and pre-code as much as possible. It saves time and costs of data handling and also one source of error: error introduced at the coding stage (Swift, 2006).

Special issues related to coding

Missing data: Missing data is probably one of the biggest data quality problems and, depending on the fraction or patterns of missing cases, can render a data set unusable. Data can be missing for a number of reasons, what is important is that the end-user knows why data are missing and this requires that one is able to distinguish between different types of missing data.

As an illustration, consider the fictional data in Table 12.3 below, with the variables ownplots: whether or not a household has agricultural plots (1 = yes), and manure: how much manure the household used (in kilograms).

Imagine that these are the reasons why data on manure use is missing

- 1002: does not own or operate any agricultural plots
- 1003: did not remember how much manure was used
- 1004: respondent refused to answer the question
- 1006: did not apply any manure, used inorganic fertilizer instead

Table 12.3 Hypothetical data to illustrate problems with missing data

<table>
<thead>
<tr>
<th>housecode</th>
<th>ownplots</th>
<th>manure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1001</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>1002</td>
<td>0</td>
<td>.</td>
</tr>
<tr>
<td>1003</td>
<td>1</td>
<td>.</td>
</tr>
<tr>
<td>1004</td>
<td>1</td>
<td>.</td>
</tr>
<tr>
<td>1005</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>1006</td>
<td>1</td>
<td>.</td>
</tr>
</tbody>
</table>
The question of manure use is not applicable for household 1002 and, therefore, this household can be excluded from any analysis on manure use. Clearly 1006 should have 0 for manure use. The other two cases are interesting in the sense that if one were to use quantity of manure used as an explanatory variable, one may wish to do something about 1003 and 1004, for example imputation. Lastly, one may be interested in understanding why respondents refuse to answer certain questions. Without codes allowing for these distinctions, it is not possible to do any of this.

Each program will often have its own codes for missing data; the commonly used ones are 99, 999, or 9999. It is okay to use these codes as long as they do not fall within the range of potential values for the variable, that is, it is not advisable to use 99 to denote missing data for age because there can be people aged 99. For PEN, we used negative numbers as codes for different types of missing data as a solution to this problem.

**Observation unit identifier (OUI):** It is increasingly common to carry out multi-level surveys. One may carry out village surveys for contextual information, household surveys for the household data and plot level surveys for agricultural production data. Moreover, under each category, multi-surveys may be conducted. As an example, to shorten the recall period, PEN chose to use quarterly surveys to collect income data.

In order to combine data from the different surveys, there needs to be a common variable and code that allows the surveys to ‘talk to each other’. This code is the observation unit identifier, a unique code assigned to each household/village and consistent across all surveys in which the village/household appears. It is advisable that this is a number for the very problems related to text data that we discussed above.

**Other:** It is common to use the code ‘other’ for all responses that fall outside known ones. However, while the ‘other’ code is useful to avoid gaps in the data, it is not helpful if a large proportion of your responses are classified as ‘other’. Imagine a table, which shows that the most important crop, contributing 50 per cent of the sector income, was ‘other’. Codes are cheap. It is better to enter as many unique answers as possible, by assigning new codes. Later, during the analysis stage, it is easy to re-classify or re-code these into fewer aggregate codes or the other code.

**Hierarchical coding:** A useful way to create codes is to think of them as being part of a hierarchy that starts with a high level category, a parent code, which is passed on to the subsequent subcategories, child codes. As an example, forests could be coded as 100, natural forests as 110 and closed natural forests as 111. These types of hierarchical codes introduce structure that results in better organization of data. In addition, because the child codes contain information about the parent codes, they give users more flexibility on the level of detail for the various analyses.
However, this must be used with some care, as codes risk becoming very long (with higher risks of errors in data entry).

**Maintenance of the codebook:** It is advisable to have one person responsible for creating, updating and maintaining the codebook. This ensures that there is no duplication of codes. It is definitely not a good idea to have codebooks developed separately with the hope that they will be merged into a single codebook.

**Coding on or off the fly:** Coding on the fly or coding as you go refers to the situation where one assigns codes concurrently with data entry. As discussed, this is inefficient and error prone. It is advisable that all responses are coded before data entry begins to ensure a smoother and less error-prone exercise.

### Data entry

The objective of data entry is to accurately transcribe data from the questionnaire to the computer and it should not be confused with data management or data analysis. This may sound obvious, however, it is not uncommon to see data entry sheets such as the one in Figure 12.2 that, in addition to the raw data, may have aggregated data, a codebook and may even include charts. In Figure 12.2, columns A and B are the raw data, D and E are aggregate data, and F–M is a chart plotting the data in D and E. Doing this is a certain recipe for data quality problems and is highly discouraged.

The conventional way to enter data is in what is referred to as a rectangular format, also called the data matrix, illustrated in Table 12.4, where columns are variables (or fields). These generally correspond with questions on the questionnaire. Rows are observations or cases, which would correspond to the observation units of the study (for example, households in a household survey or

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>4099</td>
<td>3</td>
<td>1</td>
<td>4564.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4749</td>
<td>3</td>
<td>2</td>
<td>5967.625</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3799</td>
<td>3</td>
<td>3</td>
<td>6429.231</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4816</td>
<td>3</td>
<td>4</td>
<td>6071.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7827</td>
<td>4</td>
<td>5</td>
<td>5912</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5788</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4453</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5189</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10372</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4082</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11385</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14500</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15906</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3299</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5705</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 12.2** An example of how not to proceed: Mixing raw data, aggregate data and data analysis
villages in a village survey). Cells are the data on a given variable for a specific observation unit.

In designing the system for data entry, the following should be kept in mind:

**Ease of use:** It is pointless to have a ‘world-class’ data entry system that you cannot use, as this will not only make the data entry process longer, you are also more likely to make data entry errors. That said, what may appear to be an intimidating system may turn out to be easy to use once a user has familiarized themselves with it and so it is worthwhile to try to acquaint yourself with a new data entry system and make a decision based on your experience with it.

**Restrictions:** A good data entry system should allow a user to set restrictions on what type of data is entered in a given column or cell. Examples of restrictions include:

- **Variable type:** This means text (string) or numeric. Once a variable is defined as numeric, you should not be able to enter text, however, the reverse is possible. This also means that once a variable is defined as numeric, you cannot enter ‘NA’ for not applicable and will thus need to create a code for it.

- **Range of values:** As discussed earlier, one of the advantages of working with numeric data is that you can restrict the valid entries for a variable to a range or list. As an example, if the variable ‘maristat’ (marital status) took on four values, married, single, divorced, widowed, then the list of valid entries could be defined to take on values 1, 2, 3 and 4.

- **Precision:** Without going into much detail here, there is no point allowing for decimal points when entering data on the marital status, as all entries are integer values with no decimals. However, for example, the quantities harvested should allow for decimals.

**Graphical user interface (GUI):** Most data entry programmes will have, as the underlying structure, some sort of spreadsheet that stores data in the rectangular format. A few programmes allow the user to design an interface which can ease data entry. This interface could in principle be designed to look like the

<table>
<thead>
<tr>
<th>Household</th>
<th>Age</th>
<th>Sex</th>
<th>Education</th>
</tr>
</thead>
<tbody>
<tr>
<td>1001</td>
<td>79</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1002</td>
<td>62</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1003</td>
<td>48</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>1004</td>
<td>62</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 12.4 Example of a conventional data matrix
questionnaire which then gives familiarity and therefore speeds up the data entry process.

**Which software to use?**

Word processors (such as Microsoft Word) or text editors (such as Notepad) can be used to record and store data, however, these are more suited to recording free-flowing text. It is possible, in word processors, to get the rectangular format by creating a table, but that is all it will be, a table without any of the important features discussed above. As such, word processors and text editors are not recommended for data entry. Some statistical programs such as SPSS, allow for data entry and may be used, however, the most commonly used platform for data entry is spreadsheets (for example, Microsoft Excel, Lotus, CALC). The main advantages of spreadsheets are that they have the desired rectangular format by default, their cell contents can be restricted, and users can design and add drop-down menus with pre-coded responses. Their biggest advantage is probably their familiarity. Almost all computer users have had some experience with at least one spreadsheet program and therefore find them easy to use, however, this ‘advantage’ is their Achilles heel because they are as easy to abuse and must therefore be used with great care. Stories abound of data catastrophes resulting from careless use of spreadsheets.

Arguably, the best way to enter data is to use a database. The next logical question is, what is a database and how does it differ from the spreadsheets discussed above? According to Harrington ‘there is no term more misunderstood and misused in all of business computing than database’ (2009, p4) and yet because their importance, there is no shortage of material on them. A comprehensive treatment of this material is covered in database theory, which is beyond the scope of this chapter.

For this chapter, we use the term database to refer to a mechanism that is used to store information, or data (Stephens and Plew, 2001). As with spreadsheets, databases allow users to enter, modify, delete and retrieve data, however, they have additional advantages, which include restrictions on cell contents, the capability of designing a graphical interface to ease data entry and security of data through restrictions on access. Beyond these, one of the most important attributes is the capability of using multiple but related tables to enter data. This is made possible by what is referred to as a primary key, which is the combination of one or more column values in a table that make a row of data unique within the table (Stephens and Plew, 2001). Multiple tables store data efficiently and effectively (Schulman, 2006). For more on multiple tables and the advantages of databases, see Schulman (2006).
Data checking/cleaning

‘I found mothers younger than their children. I found men who claimed to have cervical smears’ (Anonymous, 2007). It is hard to overstate the importance of data checking/cleaning and how distressing it can be. Anonymous (2007) writes that real data gives the angst of wondering whether the people who fill in forms bear any resemblance to the great mass of humanity or whether they got the dog to tick the boxes.

It is not that the enumerators and data entry clerks bear no resemblance with the great mass of humanity. Rather, this can be thought about as the classic principal–agent problem where, if you do not monitor quality and link it to pay, quality plays second fiddle to quantity (data entry clerks are usually paid per questionnaire). Data checking is a way of monitoring quality. Visual inspection is the most basic form of data checking but as the human eye can only see so much, the bulk of data checking should be done by computers, which are faster, more accurate and more thorough. Data should be checked for, among other things, duplicates, illegal entries, missing values, inconsistencies, and outliers.

**Duplicates:** As discussed above, it is important that each observation unit is uniquely identified, particularly if you work with multiple tables. This ensures that data are assigned to the right unit. In relational databases, this is achieved by the primary key which uniquely identifies each record. However, if a relation database is not used, it is important to check for that there are not duplicates in the record identifier.

**Illegal entries:** Generally, numeric variables are either discrete or continuous. Discrete variables are those that have a finite set of values, for example, the variable sex will generally take on two values, male or female (which can be coded as 0 and 1 respectively). Continuous variables are those that taken on any range of values. Both types will have illegal entries but it is much easier to identify illegal entries in discrete variables because the set of values is known apriori. Illegal entries can be checked using frequency tables, graphs or queries.

**Frequency tables**

Given that we know all the possible values a variable can take on, for example, the variable hhc_sex (sex of member) with two values, 0 and 1, a simple table of

<table>
<thead>
<tr>
<th>Sex of member</th>
<th>Frequency</th>
<th>Per cent</th>
<th>Cum.</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>579</td>
<td>53.02</td>
<td>53.02</td>
</tr>
<tr>
<td>1</td>
<td>512</td>
<td>46.89</td>
<td>99.91</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.09</td>
<td>100</td>
</tr>
<tr>
<td>Total</td>
<td>1092</td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>
frequencies will show if there are any illegal entries. In Table 12.5, we see that we have an illegal entry, 3.

**Graphs**
The same information above can be seen graphically as illustrated in Figure 12.3.

![Figure 12.3 Example of uncovering illegal data entry using graphics](image)

**Queries**
While both the table of frequency and the graph can show that you have some illegal entries, they are not efficient for at least two reasons:

i. **Unnecessary information**: When identifying illegal entries, you are not necessarily interested in the distribution of the valid entries and yet both the table of frequencies and the graph show this information.

ii. **Culprit not revealed**: While both tell us that we have an illegal entry, 3, they do not reveal which household/member has this illegal entry so you still have to go an extra step to identify the household.

It is therefore better to query the data and print out the offending observations. Queries take the general form:

1. **ACTION**: select, list, print
2. **VARIABLE**: the name of variable(s) you would like to check
3. **SOURCE**: the source of data, typically a table.
4. **CONDITION**: what you consider an error, e.g. values outside a list or range of expected values
As an example, if we wanted to see which household, and which member of the household, has the variable sex entered as 3, the Stata command and output to identify the culprit unit would be:

```
list housecode hhc_pid hhc_sex if hhc_sex < 0 1 hhc_sex > 1
```

<table>
<thead>
<tr>
<th>Housecode</th>
<th>hhc_pid</th>
<th>hhc_sex</th>
</tr>
</thead>
<tbody>
<tr>
<td>317</td>
<td>7</td>
<td>3</td>
</tr>
</tbody>
</table>

Figure 12.4 Example of Stata query to find illegal data entry

**Missing data:** As mentioned above, gaps should be minimized and codes for missing data should be used. Nonetheless, during data entry, some cells may be skipped in error. As with illegal entries, a simple table of frequencies can show missing data. However, a query will identify the specific offending observation.

**Inconsistencies:** It is common for questionnaires to have interrelated questions, where one question leads to another or where some questions depend on the response given to previous questions. See Table 12.6 for an example from PEN’s A1 survey (see Chapter 1).

You should check that all migrant households (who answer no (0) in question 3, Table 12.6) report how many years they have lived in the village.

**Outliers:** Most data, particularly cross-sectional survey data, contain unusual observations, unusual in the sense that they are inconsistent with the bulk of the data being either too large or too small. Such observations are commonly referred to as outliers.

The most used technique for identifying outliers is the boxplot proposed by Tukey (1977). This method uses ‘fences’ as criteria to label an observation an outlier with values less or greater than the lower upper fences respectively classified as outliers. The fences are defined as \( L_f = P(25) - k \cdot IQR \) and \( U_f = P(75) + k \cdot IQR \) for the lower and upper fence respectively, and where \( P(25) \) and \( P(75) \) are the 25th and 75th percentiles, \( IQR \) is the interquartile range \( (P(75) - P(25)) \) and \( k \) is an arbitrary number. In general, a value is a mild outlier if it falls outside (above and below) the fences given by \( k = 1.5 \) and an extreme outlier when \( k = 3 \).

However, Hubert and Vandervieren (2008) note that when the data are skewed, usually too many points are classified as outliers. We found this to be a real and big problem in the PEN data set, for example, price data tended have spikes/lumps because of very little variation in product prices for any given product unit combination. Our data were also skewed, for example, data on agricultural input use tended to be skewed to the left with the bulk of the households using none or very small quantities. Because of this, we wrote a Stata program – `obsofint` – (Buis and Babigumira, 2010), an implementation
of Tukey’s fences combined with the rule of huge error outlier labelling method with some modifications to account for skewed or lumped data.

**Conclusions**

A good quality data set is a basis for useful information that informs policy and aids decision-making. Practically, a good quality data set lends itself to multiple visits by the same or different users, likely yielding new insights with each visit. In contrast, bad quality data may lead to wrong interventions, is frustrating to use, and may ultimately be rendered unusable and discarded, which would be a highly undesirable outcome given the costs and time spent collecting the data.

While data entry and checking are further downstream in the research process, they should not be dealt with in an ad hoc manner, rather, they should be incorporated at the outset so that they can inform and be informed by the research process. Do not wait until the survey is done to begin thinking about data entry and storage. As you design the questionnaire, think about how data will be entered and stored and, if possible, concurrently design your data entry system.

It is better to enter numerical rather than text data so try to code all categorical responses prior to data entry. One person should maintain the codebook and existing codes should never be overwritten. Instead, add new codes and do not worry about the size of the codebook. Try not to create aggregate codes at the coding stage as you can easily do this when the data have been entered. Remember that aggregating data comes at the cost of loss of detail. You can aggregate codes but you cannot disaggregate them once the data have been entered.

In trying to get good quality data, the winning strategy is ‘prepare and prevent’. Think about the potential sources of error and try to put in place mechanisms to minimize the errors. Lastly, leave a paper trail. Documentation improves data quality so record as much as you can while you still remember.

**Table 12.6** An example from PEN’s A1 survey

| 3. Was the household head born in this village? If ‘yes’, go to 5. | (1-0) |
| 4. If ‘no’: how long has the household head lived in the village? | years |
Key messages

- Use codes instead of text, if possible.
- Data entry is about entering data and should not be mixed with data analysis.
- Ensure that all observation units are assigned unique codes.
- Don’t wait until the end of all field work to start thinking about data entry.

Notes

1. See www.multiple-imputation.com for some references.

References


StataCorp (2009) *Stata Statistical Software: Release 11*, StataCorp LP, College Station, TX

