Gayle, Vernon, and Paul Lambert. "Getting started." *What is Quantitative Longitudinal Data Analysis?*. London: Bloomsbury Academic, 2018. 119–128. *Bloomsbury Collections*. Web. 26 Apr. 2023. <<http://dx.doi.org/10.5040/9781472545268.0014>>.

Downloaded from Bloomsbury Collections, www.bloomsburycollections.com, 26 April 2023, 11:42 UTC.

Copyright © Vernon Gayle and Paul Lambert 2018. You may share this work for non-commercial purposes only, provided you give attribution to the copyright holder and the publisher, and provide a link to the Creative Commons licence.

7 Getting started

The preceding chapters have set out ‘what is’ longitudinal data analysis. This final chapter is more practical and outlines ‘how to begin to undertake’ longitudinal data analysis. It is envisioned as an aid to making a smooth departure on the journey of longitudinal research, and it is not intended to be a comprehensive guide to undertaking data analyses.

It is infeasible to attempt longitudinal social science data analysis without using data analysis software. Several statistical data analysis packages are available to social science researchers. The three that are currently most prevalent are SPSS, Stata and R. We advocate using Stata for longitudinal data analysis, and a justification for this preference is outlined in Gayle and Lambert (2017). Researchers are of course free to use whichever software they prefer.

The workflow

Software can be operated in different ways, but the complexity of longitudinal datasets means that nearly all highly skilled researchers write out software commands using a syntactical or programming format, which in turn forms an audit trail. The audit trail is a central aspect of the workflow, which includes planning, organizing, executing and documenting analyses.

We advise all readers to consult Gayle and Lambert (2017), which provides detailed prescriptions on the social science data analysis workflow, and we are fairly certain that it will help most researchers regardless of how much experience they have analysing social survey data. A more wide-ranging account of the workflow is set out by Long (2009), and in our view Long is the definitive text on this subject. In Gayle and Lambert (2017) we treat a number of highly important practical issues that are usually overlooked in data analysis texts. These topics include organizing directory structures, adopting file-naming conventions, keeping track of

files (i.e. version control), managing variables, and structuring and documenting software syntax files.

Documentation is fundamental to achieving a workflow that is accurate, efficient and transparent, and which facilitates reproduction. Long (2009) posits Long's law, which states that it is easier to document today than it is tomorrow. He further states that there are two corollaries. First, nobody likes to write documentation. Second, nobody ever regrets writing documentation. We suggest that in the history of social survey data analysis, no one has ever said 'This work is too well documented'. We encourage longitudinal data analysts to make frequent notes and comments because these will be the backbone of documentation.

Analysing longitudinal data without a planned and organized workflow can be compared to drinking and driving. In both situations, no matter how careful you are, it is still highly likely to end in a wreck!¹ Therefore, just like drinking and driving, we strongly warn against not having a systematic workflow and in our view no researcher should ever undertake any serious survey data analyses without a planned and organized workflow.

Data enabling

Social surveys almost always require some preparatory work before analyses can be undertaken. We use the term 'data enabling' to describe this phase of the workflow. In reality, even appropriately curated social science datasets will require some work to be undertaken to enable data analyses, even if only recoding some variables or coding some missing values.

Longitudinal datasets are almost never provided in a form that renders them immediately ready for comprehensive analyses. Social scientists face many practical difficulties when enabling longitudinal data for analyses. The most obvious challenge is linking up individuals at multiple time points. Less obvious are the difficulties associated with matching individuals with others in the study, for example spouses, children, siblings or other household sharers. Many research questions also require the addition of

---

¹ We are grateful to Professor Philip Stark, University of California Berkeley, for this useful and clear analogy.

important contextual information, for example about the household or locality, which can also prove challenging.

A substantial but often under-appreciated aspect of working with longitudinal datasets is the challenge of constructing appropriate measures which aid comparability over time or which suitably reflect social changes. There are often a number of different measures (e.g. different socio-economic classifications) in the dataset that could be used, and often metadata are also supplied with the dataset. A weakness of some analyses is that they do not take full advantage of the richer data resources available in the survey because they are restricted to exploiting only the most readily available information.

Over the course of our careers we have learnt the important lesson that the process of data enabling usually takes five times as long as initially anticipated. Data enabling tasks that researchers initially think will take an hour usually take five hours, and similarly tasks that we think will take a day end up taking five days. This is an important lesson to internalize, and our advice is to try to ensure that you build in adequate time for enabling your data and try to remember that data enabling tasks are usually very time consuming. Our graduate students refer to the idea of ‘five times’ as the ‘Gayle-Lambert constant’. We prefer to think of it as a guideline because there will undoubtedly be social scientists with better programming skills than ours, who can work faster and program more efficiently.

Using Stata

In our view the software Stata is particularly well suited for the combined tasks of data enabling and data analysis that characterize longitudinal social science data analysis. There are many resources aimed at providing Stata training. We recommend Kohler and Kreuter (2012) and Mehmetoglu and Jakobsen (2016) because they are well-organized and informative texts. Some students prefer Pevalin and Robson (2009). We also recommend Treiman (2009), which is an excellent text covering a wider range of advanced topics and issues that researchers are likely to encounter. We enthusiastically promote the Institute for Digital Research and Education (IDRE), University of California, Los Angeles, web resources² and draw

² See <https://idre.ucla.edu/> (accessed 6 April 2016).

attention to their Stata pages³ and very helpful annotated outputs pages which contain example programs and outputs with footnotes explaining the meaning of information, measures and statistical tests reported in Stata outputs.⁴

As new versions of Stata emerge, commands tend to remain consistent, but occasionally there are changes and some features are no longer supported. The *version* command is a valuable tool in Stata. It ensures that older material (which is sometimes outdated) can be run by newer versions of Stata. For example, the command *version 12* will make Stata version 14 behave as if it were Stata version 12. It is possible that in the future some of the material provided in this book will be incompatible with future versions of Stata and this command may be useful. If you are working in a collaborative project with colleagues that have older versions of Stata, it is often good practice to save files using Stata's *saveold* command.

A benefit of Stata is that new commands and functions are often developed, which can be incorporated into your current version of Stata. It is possible to acquire and manage downloads from the internet using the command *net*. The *findit* command can be used to search the Stata site and other sites for information. For example, imagine that you heard about a program to draw graphs using quasi-variance estimates. Using the syntax *findit qvgraph* would lead you to the module written by Aspen Chen of the University of Connecticut.

Many new packages are deposited at the Statistical Software Components (SSC) archive, which is sometimes called the Boston College Archive and is administered by <http://repec.org>.⁵ The SSC archive has become the premier Stata download site for user-written software and also archives the proceedings of Stata Users Group meetings and conferences. Programs can be downloaded from the SSC archive using the syntax *ssc install* followed by the new program's name. Readers who do not have administrative access to Stata (e.g. when working on their university network) may first require permission to download packages.

---

³ See <http://www.ats.ucla.edu/stat/stata/> (accessed 6 April 2016).

⁴ See <http://www.ats.ucla.edu/stat/AnnotatedOutput/> (accessed 6 April 2016).

⁵ Accessed (12 June 2016).

An alternative approach may be to set up an area locally where you have write access (e.g. `c:\temp`) and then use a variant of the following Stata syntax:

```
global path10 "c:\temp\"
capture mkdir $path10\stata
capture mkdir $path10\stata\ado
adopath + $path10\stata\ado
net set ado $path10\stata\ado
```

You can test this by installing the *estout* package from SSC:

```
ssc install estout
```

Help on this new package should then be available:

```
help estout
```

Longitudinal data structures in Stata

Most researchers who are used to working with social surveys will be familiar with datasets that are ‘variable by case’ matrices. The dataset will usually be composed of many rows, each containing information on a single unit (e.g. an individual). It will also be composed of a set of columns. Each column will usually contain information for a single variable. A simple example is illustrated in Figure 48.

Sometimes longitudinal datasets are constructed in ‘wide format’. In wide format each row of data represents a single unit, such as a respondent, but some or all of the variables are measures about the unit from different time points. A common strategy is to label relevant variables with

id	Age	female	working_hours
001	20	0	37
002	30	1	39
003	40	1	45

Figure 48 An example of variable by case matrix

a prefix or suffix that indicates the time point that it relates to. A simple example is illustrated in Figure 49. As we discussed earlier there are some longitudinal techniques that require the data to be organized in wide format (e.g. some duration models).

There are other data analysis techniques, for example the panel models that we introduced in Chapter 5, that require data to be organized in ‘long format’. In long format, units such as individuals can contribute multiple rows of data, each for a different time period. For example, the wide format dataset that was outlined in Figure 49 can be reorganized into long format, as displayed in Figure 50. The columns of data are no longer specific to different time periods; however, it is important that the dataset contains a variable indicating which time period the row relates to. In Figure 50, this information can be found in the variables *wave* and *year*, which represent the survey wave and the calendar time.

Many longitudinal studies provide information in different data files for different time periods. For example, the main body of the BHPS is supplied as a collection of 18 files for the data collected from adult respondents in

id	female	age_1971	age_1972	age_1973	work_hours_1971	work_hours_1972	work_hours_1973
001	0	20	21	22	37	40	35
002	1	30	31	32	39	40	35
003	1	40	41	42	45	45	15

Figure 49 An example of a wide format dataset

id	wave	year	female	age	work_hours
001	1	1970	0	20	37
001	2	1971	0	21	40
001	3	1972	0	22	35
002	1	1970	1	30	39
002	2	1971	1	31	40
002	3	1972	1	32	35
003	1	1970	1	40	45
003	2	1971	1	41	45
003	3	1972	1	42	15

Figure 50 An example of a long format dataset

each of the 18 years of the survey (known as wave ‘a’ through to wave ‘r’). Usually, if a long format dataset is required, the researcher will open up each file from each year, extract the data that they need, and then add the files together ‘on top of each other’. In Stata, this involves using the *append* command and taking care to store information on individual identifiers and time periods.

If a wide format dataset is required, the researcher opens up each file from each year and extracts relevant information, but then matches together different data from different years based upon shared individual identifier variables (e.g. by using Stata’s *merge* command). Figure 51 illustrates some typical Stata syntax that may be used for this process for a scenario where we use the first three years of individual-level data from the BHPS and collate data on current marital status (the variable *mastat*). The syntax takes advantage of some standardized file- and variable-naming conventions used in the BHPS. For instance, the prefix letter for a file indicates the year of the data and the prefix letter for a variable within a file similarly indicates the year that the data applies to. Other large-scale social science datasets often feature similar conventions for file and variable names.

Some researchers make use of the Stata command *reshape*, which is a convenient command that is specifically designed to convert longitudinal datasets from wide format to long format and vice versa. There are several useful examples using the *reshape* command in the Stata manual and there is an extended account in Chapter 8 of Mitchell (2010). We generally do not use the *reshape* command in our own work as we tend not to have to move frequently between the two different formats. We have observed that novice data analysts sometimes get into a tangle using the *reshape* command. This is because there is a small risk of losing track of transformations when moving between data formats. A little more work is required but we believe that in general it is better to construct datasets in either format through a more deliberate (and therefore transparent) set of syntax commands.

It is important to stress that in most research scenarios the sequence of commands used to bring together data from different time periods will be much more complicated than the illustration in Figure 51. There will typically be activities associated with coding, harmonizing and documenting variables and dealing with missing data that will take considerable thought, time and effort. There are few situations in data enabling where the case for careful workflow documentation and the use of syntax and

Figure 51 Sample Stata syntax illustrating the construction of long and wide format datasets using the BHPS (and images of the data formats that are generated)

command files is more compelling. A project will run more smoothly if there is a clear and reproducible record of syntax and the steps taken in opening files and processing data from different sources. A longitudinal data analysis project that does not have an organized and systematic workflow with a clear audit trail will be storing up trouble for a later date!

Last words

Oliveira and Stewart (2006) conclude that if your program (in our case the syntax file) is not correct nothing else matters, for without correctness

we cannot expect any useful results. Long (2009) reminds us that there is an unavoidable tension between undertaking work carefully and completing work, but he also adds that the production of incorrect results injures both the researcher and the research field.

Using software syntax is central to a well-organized longitudinal data analysis workflow and much of the time spent analysing data involves undertaking repetitive tasks. As Long (2009) helpfully reminds readers, repetition invites errors, and automation which can be achieved through suitably programming software is both faster and less error prone. Developing good data enabling skills pays important dividends because these skills allow researchers to take fuller advantage of the richer data resources available. Good data enabling skills will free researchers from the shackles of having to analyse only the most readily available information. The most obvious payoff for having an effective workflow is that accurate results can be produced systematically and efficiently. Academic life is measured by outputs and structured by deadlines, and an organized workflow can make an essential contribution to making progress.

Data citations

- Croxford, L., Iannelli, C., Shapira, M. (2007). *Youth Cohort Time Series for England, Wales and Scotland, 1984-2002*. [data collection]. UK Data Service. SN: 5765, <http://dx.doi.org/10.5255/UKDA-SN-5765-1>.
- University of Essex. Institute for Social and Economic Research. (2010). *British Household Panel Survey: Waves 1-18, 1991-2009*. [data collection]. 7th Edition. UK Data Service. SN: 5151, <http://dx.doi.org/10.5255/UKDA-SN-5151-1>.
- University of Essex. Institute for Social and Economic Research, NatCen Social Research. (2015). *Understanding Society: Waves 1-5, 2009-2014*. [data collection]. 7th Edition. UK Data Service. SN: 6614, <http://dx.doi.org/10.5255/UKDA-SN-6614-7>.