BY THE END OF THIS 90 MINUTE CHUNK OF TIME I WILL HAVE A COMPLETED SCRIPT WHICH SHOWS HOW TO USE AUTOMATED DATA MANAGEMENT TO EXPLORE DEMOGRAPHIC DATA, AND SOME MARKDOWN DOCUMENTS WHICH EXPLAIN SOME OF THE KEY ARGUMENTS AND IDEAS.

THE KEY THING MISSING FROM THE SCRIPTS IS A 3D DATA VISUALISATION OR LEXIS SURFACE.

**Introduction: Why automate?**

Welcome to a one day course on computational social science in R. This course will provide some examples of automating processes involving (fairly) large amounts of data, which we’ll run and discuss over the course of the day.

The data we will use come from the Human Fertility Database (HFD) and the Human Mortality Database (HMD), both of which are run by the Max Planck Institute for Demographic Research along with other partners. As the names suggest, they both comprise demographic data, which I think of as ‘old big data’. The HFD is around 0.2Gb in size; and the HMD around 1.3Gb. Compared with some of the data sources currently being generated and analysed – such as those which are generated when customers use a website, or when people use hardware and apps to track their activity on a second-by-second basis - these are tiny amounts of data; they are also much less ‘messy’, and require less data tidying and processing, than such forms of data. However, they are large by the standards of much of the academic social sciences, freely available, and more importantly substantively interesting and important. I therefore hope some of the analyses and results presented here will interest, engage and surprise, even though they are being used pedagogically in this case, to introduce a broader pattern and approach to data process automation.

Process automation is important when there is a need to do the same thing, or almost the same thing, many times, quickly, and without substantial risk of human error. R is a statistical programming language, and like other programming languages allows many processes to be automated by specifying general sequences of operations to be performed to data in the form of functions. Functions take a particular type of input, ‘work on’ (process) that input in a particular way, then produce a particular type of output. If you can specify the process clearly enough, and in a general enough way, then the same function can be re-used many times, each time processing a different input in the same way. When there are many dozens, hundreds or thousands of inputs that all need the same ‘work’ doing to them, then using a function to process multiple inputs can save minutes, hours, days or even weeks of your time.

The vague but useful mantra of data science is often described as ‘turning data into knowledge’. The input is ‘data’ and the output is ‘knowledge’ (or ‘value’, or ‘information’, or ‘insight’), and ‘data science’ is the magical pipeline linking input to output. However, (to paraphrase Arthur C Clarke) data science isn’t really magic, it’s just an advanced data processing technology, made up of many other smaller and simpler technologies. Each of these simpler technologies can also be thought of as a ‘function’, with a specific input, process, and output. These functions chain together, with the output from one function becoming the input to another. Often, the chain of functions isn’t even linear: the output from a later function in the chain can feed back to form an input to an earlier chain.

The technology isn’t even always fully automated, and nor should it be. The ghost in the machine is you, the researcher, making important analytical decisions about how to get more knowledge and insight from the data, given what you already know. As you work with the data, you learn more, perhaps leading you to choose to rebuild the machine to produce new outputs, and generate new knowledge. Being the ghost in the machine, the wizard behind the curtains, can be fun. But only if the machine works smoothly and efficiently. Computational approaches to social science allow you to spend more time being a social scientist, running the machine, by ensuring you spend less time being part of the machine. For example, if the output from one process needs to be carried by wheelbarrow to the input to another process, the guy with the wheelbarrow is part of the machine. If dozens of different input pipes need to be connected manually to the same unit, the guy constantly screwing and unscrewing pipes is part of the machine. Computational social science aims to make you and your work much less robotic, by systematically removing the need for you to be your own knowledge labourer in your own knowledge-making system. However, doing this requires learning a number of methods and patterns for working holistically and systematically with data challenges. This one day course will introduce some of these key patterns.

**Key stages in the data-to-value chain**

There are often a great many intermediate steps involved, but often the core stages involved in working with large amounts of data are:

1. Inputting ‘raw’ data
2. Producing ‘tidy’ data
3. Initial exploratory analyses
4. Producing summary statistics and visualisations for each of the inputs
5. Producing final results and outputs

The first, second, and forth stage can all be automated, and doing this tends both to save time for stages three and five, and often to allow stages three and five to be performed more quickly and effectively. In the scripts used on this course, stage one and two are largely combined, but it is important to think about them as distinct activities. The same broad tasks will be performed using both the HFD and HMD, in order to make the broader pattern and process easier to identify.