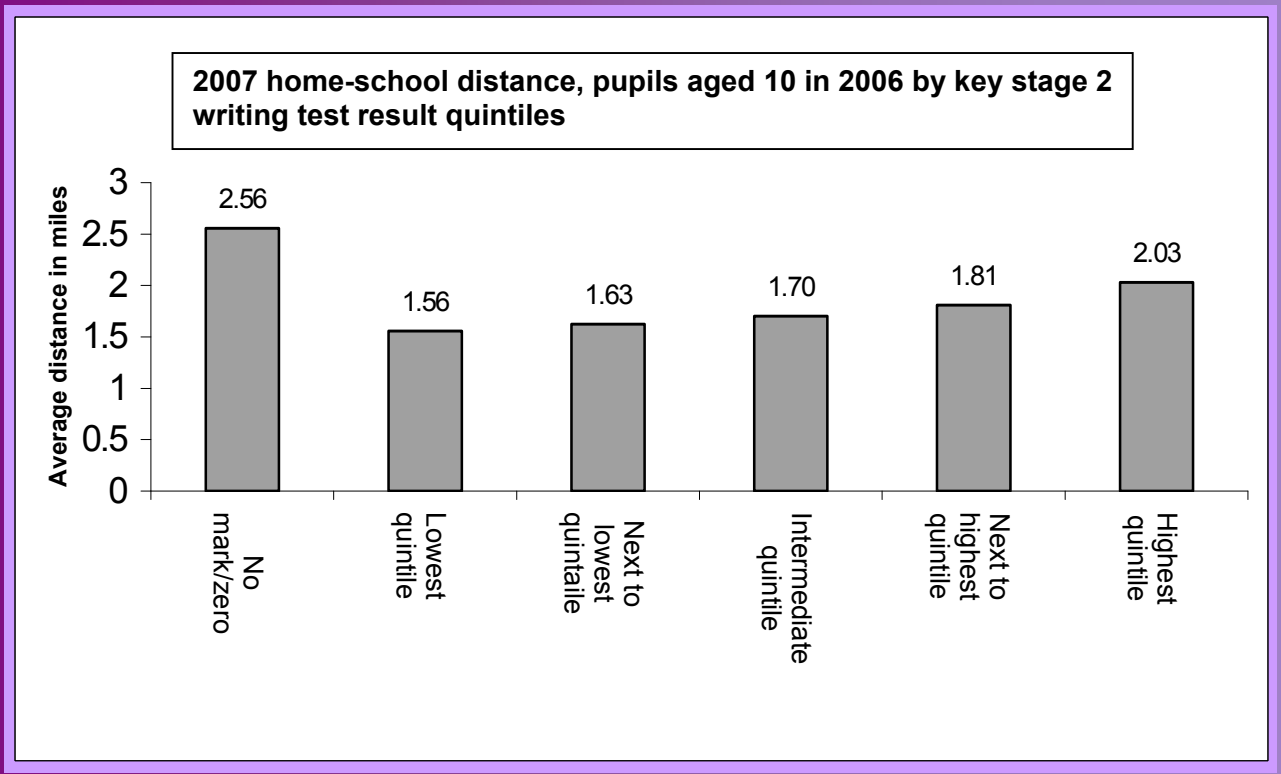
GREATERLONDONAUTHORITY

Data Management and Analysis Group

A Threshold Guide to Readying Data for Analysis in SPSS:

Quantitative analysis and Elizabeth David, the English National Pupil Dataset, a hidden variable, the Julian calendar and more



DMAG Education Guide 2009-1
September 2009
ISSN 1479-7879

D.C. Ewens

DMAG Education Guide 2009 - 1

September 2009

A threshold guide to readying data for analysis in SPSS:

Quantitative analysis and Elizabeth David, the English National Pupil Dataset, a hidden variable, the Julian calendar and more

David Ewens
Data Management and Analysis Group
Greater London Authority
2nd Floor
City Hall
The Queen's Walk
London SE1 2AA

Tel: 020 7983 4656
Email: david.ewens@london.gov.uk

Acknowledgements

In its pre-Windows days, SPSS could be difficult for the newcomer to get to grips with, particularly if there were no workshops available and the would-be user had no background in Fortran. I remain appreciative of the help Pauline Booker, of the then School of Social Sciences at the Polytechnic of Central London, gave me during my earliest days with the software. A number of other people have made useful suggestions since, including Sean Hayes during his time with Hammersmith and Fulham's Education Department, Rachel Leeser of the Social Exclusion Team in the GLA's Data Management and Analysis Group (DMAG), and Richard Cameron of DMAG's National Census Team. The Guide is not a mainstream DMAG research and statistics report and is, to that extent, a somewhat heretical departure from the Group norm. However, the Guide is prompted by research necessity rather than whimsy (or even heresy) and I am appreciative of the time John Hollis and Rob Lewis in DMAG allowed for it to be written.

Further support

Advice on the next steps that might be taken by those new to readying data for analysis in SPSS is given in the concluding Section of the Guide. Regrettably, the pressure of research and statistical analysis at City Hall means that I cannot undertake to respond to requests for further advice or support.

DMAG Education Guide 2009 - 1

September 2009

A threshold guide to readying data for analysis in SPSS: Quantitative analysis and Elizabeth David, the English National Pupil Dataset, a hidden variable, the Julian calendar and more

Contents

1. Introduction and key issues

1.1 Why a DMAG Education guide to readying data for analysis in SPSS?

The Guide provides an introduction to a number of SPSS procedures that are useful in organising and readying data for analysis. This is a largely, though not entirely, neglected element in the literature on SPSS. It is not a general introduction to SPSS, or an introduction to statistical analysis in SPSS; a large number of introductory guides to the latter are already available. Indeed it assumes that the reader has experience of statistical analysis in SPSS, and is at home with its menus, file naming procedures and so on.

That said, the Greater London Authority's Data Management and Analysis Group (DMAG) exists to help provide the GLA with its evidence base, and output is mainly in the form of published Research Briefings. A threshold guide to readying data for analysis in SPSS is not a Research Briefing. Since there are no plaudits for DMAG's Education Team in writing such a guide, and since there is a risk that producing a computing guide may give a misleading impression of what the Team does, why write one?

The short answer is not so much that there is a title missing from the bookshelves (the researcher's 'gap in the literature' reason for writing) but rather that an introductory, threshold guide is needed. I will illustrate that point with 'a real world' example.

A grant was available for work in the DMAG Education Team on the impact of multiple social disadvantage and educational attainment. This was to be part of a 'making the case for London' project with, potentially, material benefits for schools. The grant was to meet the cost of researcher who would work on the first stage of the project, which involved readying data in large datasets for analysis in SPSS, using procedures of the type introduced in the Guide. This was an essential, and time-consuming, first step in the overall project. The time frame for the project meant that there was no time for training someone in the skills needed; the researcher needed to have those skills at the outset.

Despite valiant efforts by a several colleagues in City Hall, in a number of universities and in London boroughs, we could not find an individual who was available and who knew how to carry out that work in the time available. The project was stalled. A fuller answer to the 'why write a guide to readying data' question is that there is a skill gap, which blocked an important project in City Hall, and that skill gap is likely to have been experienced by others elsewhere.

While this skills gap was, to say the least, inconvenient, it may not be realistic to expect formal and informal research and research training in universities to be geared towards those SPSS procedures which are useful in readying data for analysis. Taught courses in statistical analysis may use SPSS for the computations involved but, unsurprisingly, will focus on statistical analysis. Likewise, the very large number of texts written on SPSS is, far more often than not, aimed at those taking university courses in statistics and, again unsurprisingly, focus on statistical analysis. Additionally, datasets used in higher education *can* be small, and are often purpose built and coded for a taught course or to deal with a comparatively narrow set of research questions. The issues raised in this Guide *may* not be particularly widespread in universities. Market research has also been a growth area for SPSS, but private sector companies are not widely noted for writing 'how to' guides for their competitors.

Unfortunately (or not) data in the world beyond the lecture theatre and seminar room are often not ready prepared for whatever analysis an individual researcher may have in mind, and that is part of the reason why the SPSS procedures discussed in the Guide can be so useful.

Part of the issue involves the use of secondary datasets. Some extremely important secondary datasets already exist, and a number of Web links to agencies dealing with, in this instance, longitudinal data are given in the concluding section to the Guide. These are at least two types of pre-existing 'secondary' datasets. Researchers can deposit, with UK Data Archives, datasets previously constructed for their own research, and a presentation entitled 'Using and Archiving Research Data' is available at the time of writing at
www.esds.ac.uk/news/eventsdocs/birkbecknov08.ppt

Other data have been collected either to meet administrative need in an organisation, such as the payroll section of a private company, or by the state (or both). These are sometimes, but not always, referred to as 'administrative datasets', This is an unhelpful term, which does not help prepare the researcher for the range of different types of data that may be encountered, or the different forms in which it can be held. Certainly the government datasets referred to in the presentation above are referred to as government datasets rather than as administrative datasets. As a rule of thumb, refer to a dataset by its actual name, by the field or topic it covers or by the level of data involved. This is both a matter of accuracy

and, as Section 2.2 indicates, can be a matter of building good working relationships with others rather than loosing that opportunity.

At a more specific level, the researcher schooled only in SPSS may assume that data are 'normally' held in a single flat file; that is, where everything is held in one spreadsheet like block of data. Datasets are not necessarily like that, as this link to CAMSIS-CHER confirms http://www.camsis.stir.ac.uk/cher.html. Further, there is a there is a considerable body of data 'out there' which is held in software which is specifically designed to work with multiple, rather than single, files. These are relational databases, in which information in one file can link to information in another.

Characteristically, relational databases are far less demanding than SPSS on computing capacity. Rather than requiring that a whole flat file be open to run a simple bivariate equation, relational databases 'call up' just those variables needed in, for example a two-way tabulation of data. Those variables can be drawn from different datasets simultaneously, which can also ease demand on computing capacity. One payoff of this is that relational databases are much happier than SPSS in working with text (string) variables, including long string variables.

While relational databases lack the advanced statistical techniques available in SPSS (unless the user writes the programs involved, as Norman Nye and associates originally did with SPSS) they provide excellent facilities for producing arithmetic tables, and lists of data. They are widely used in government, in national services such as education and health, and in business. Relational databases are major repositories of information.

The opportunity relational databases present is one of access to data, and more information will be held in relational databases than in 'single flat file' statistics packages. Two challenges, at least for researchers using SPSS, are to bring that data together in a single flat file from multiple files, and to replace string data with numeric equivalents, which in some cases will need value labels to make them meaningful and open to statistical analysis. The Guide introduces some of the procedures in SPSS which are useful in each of these tasks.

A third challenge can be experienced by any researcher working with secondary datasets. A secondary dataset will not necessarily have been created with the researcher's current exact aims in mind. What may be an acceptable missing value in one person's database, may be an unacceptable error in someone else's. Variables that the researcher would wish included may simply not be there. Secondary datasets will need to be checked for missing data, derived variables may need to be created, and ecological variables may need to be added from other datasets before analysis can proceed. The Guide also covers each of these issues.

With a limited number of exceptions, the Guide illustrates SPSS procedures with actual work in London's City Hall on pupil level data, based on records from the (English) National Pupil Dataset (NPD). NPD records are held in a relational database warehouse, where the main aim is to produce tabulations and summaries of data, rather than to carry out advanced statistical analysis. The relational database origins of the extracts Department for Children Schools and Families (DCSF) releases is reflected in the number of files involved, and by the preponderance of text records over numeric records. A repeated theme in the Guide is that readying data for analysis take place in a context, and Section 2 points to two issues which can be thrown into sharp relief in work with pupil level data.

The Guide is aimed in part at anyone who is new to any of the procedures covered here. However, the key audience is those who lead research teams. Where there is a skill gap in readying data for analysis in SPSS, it will need to be closed. In turn this means that appropriate support will need to be in place, *preceded by interview arrangements which assess whether candidates' would benefit from that support*.

Hopefully, the present Guide will shorten the learning curve, and save time for everyone, including research leaders. However time for learning will still be need to be allowed for. If the author's experience is anything to go by, newly appointed research staff, who have already used SPSS for statistical analysis but are not familiar with the procedures described here, will need up to six months on the job, as opposed to on the training course, experience to internalise the procedures referred to in the Guide.

It should go without saying that the Guide is particularly aimed at those who have used SPSS in work with pre-coded datasets, and are now ready to move on.

1.2 A day in the life of a threshold guide

The Guide is a threshold guide, and what I mean by that is explained in the following experience. During the 1990's I worked in a London local education authority (LEA), a branch of local government which had responsibilities for maintained (state) schools amongst its portfolio of duties. My role at the time was in Education

Research and Statistics, and in 1992, we were able to establish an annual pupil survey. The survey took place on a given date in November, and collected electronic records for all pupils in all mainstream primary and secondary schools, who were on roll on the census date, or who had been on roll at any point in the previous 12 months

I had intended to use SPSS to analyse the data but, for purely local reasons, that software was not available at a key point in time. On the other hand dBase 111+, a major database package of the time, was available for immediate use and we turned to that to ensure that information on pupil mobility, special educational needs, mother tongue and on a longitudinal view of pupil fluency in English where that was an additional language, would be available to meet pre-existing deadlines.

There were two problems. I had not heard of dBase 111+ until then, and certainly had no idea of how to use it. Additionally, as a total novice, the manual appeared to me to be completely incomprehensible. Fortunately, what was then Dillon's University bookshop in Bloomsbury had a copy of Alan Simpson's step-by-step introduction to basic features of dBase 111+. This provided enough of a foundation to help me help myself. It helped me, as it were, across a threshold to the point where I could develop further skills myself (including working out what to make of the manual).

The full range of information needed in the local authority was available to deadline, and the experience provided a very forceful lesson in the potential value of threshold guides. As a threshold guide, what follows does not aim to provide an introduction to 'everything', but it does set out to provide enough of an introduction for readers to be able to go on help themselves as far as readying data for analysis in SPSS is concerned.

The Guide takes a step-by-step approach, on the premise that it will be easier for readers new to the procedures involved to take that route, rather than being faced with the need to 'unpack' more complex procedures from the outset. The Guide is cumulative, and Sections are not self-contained. What comes later tends to rest on what came before, and the Guide needs to be read from the beginning. It was not written as a manual where each Section is self-contained, and readers may well struggle if they attempt to use the Guide in that way. As said above, the Guide assumes that the reader is already at home with statistical analysis in SPSS and with the SPSS window. The worked examples in the Guide are based on SPSS 14; other versions can differ in matters of detail in the procedures outlined here.

Readying data for analysis can take up a substantial proportion of the time available for a research project, and there are approximately 25 references in the Guide to the potentially (very) time-consuming nature of that work. Just how much time is required will come as a surprise to policy makers and to researchers using, what I refer to later and only slightly tongue in cheek as 'oven ready' datasets. Research Team leaders may well need to support policy makers in coming to an understanding the amount of time readying data for analysis can take.

However, seen in proper perspective, that is in terms of the issues readying data raises, it is just one, comparatively small, element in the world of research (more of this in Section 2). Nonetheless, it involves choices that can sharpen, add, lose or distort vital information, and which can advance or delay a project. Fortunately, while there are many different ways of getting that exercise wrong, there are a number of straightforward procedures in SPSS that help in getting it right.

The issues to be covered are indicated in the Contents section. In broad terms the issues are

- the implications for demands on computing capacity
- the addition of variables, including ecological variables, from other datasets
- creating derived variables for analytical purposes
- dealing with missing data and data quality checks
- arrangements for data security and confidentiality

1.3 Computing Capacity

The NPD is a large dataset, with approximately 7.5 million individual pupil records obtained from a combination of pupil assessments files and from what was originally called the Pupil Level Annual Schools Census (PLASC), which became the January Annual Schools Census (ASC), and is now a termly School Census (SC). The size and contents of the NPD have implications for computing capacity, particularly if data for more than one year are linked together. This gives the research analyst an incentive to organise data in a way that reduces demand on computing capacity. Much of the data from the NPD are in the form of coded string (text) variables. As a statistics package, SPSS prefers to work with numeric data and, as a rule of thumb, replacing string variables with numeric equivalents will reduce the demand on computing capacity. In some instances SPSS will, in any event, not 'recognise' string variables. The research analyst will also need to be economical with the number of variables used (while ensuring that any variable

that has been deleted can be reinstated if need be).

1.4 Labelling Datasets for Descriptive Clarity

The NPD has a large number of cases (pupil records) and a large number of variables. These can be increased by the addition of variables from other datasets, and through the creation of derived variables. The merged 2002, 2003, 2004 and 2005 London Pupil Datasets has over 1,200 variables. With that number of variables, some form of labelling will be needed if the research analyst is not to lose the plot.

Additionally, some of the string variables employ a large number of non-numeric codes. The 2007 January School Census (SC), for example, contains preliminary records of languages spoken by pupils. There are more than 300 codes, including WOL, PNJM and KURM. Their meaning is not self-evident, and the list may well be unknown to the individuals with whom the research analyst is attempting to communicate. Descriptive clarity is needed not only by research analysts to avoid losing track of data in large datasets, but also in communication with the research analyst's audience. Fortunately, there is a procedure in SPSS which can be used to label numeric equivalents of string variables using a separate 'lookup' table.

Lookup tables of that type will need to be created by the research analyst, and will therefore involve effort in proportion to the number of codes involved. A language lookup table, with 300+ codes, will take time to create, and there may well be a point, in terms of the number of codes involved, where an alternative approach is needed. For example, a research analyst may need to provide policy makers with analyses of pupil attainment by ward of residence. (The SC does not contain a record of pupil home ward but, depending on the data released by DCSF, it may nonetheless be possible to add a record of home ward to the SC file.) There are more than 7,000 wards in England and, by comparison, creating a language lookup is a modest exercise. Fortunately, depending on the data available, there is an SPSS procedure that can speed up the creation of a numeric and labelled version of pupil home ward.

1.5 Adding ecological and other variables

As a dataset dealing with a national population, and now updated three times a year, the NPD has considerable potential, and this can be increased when data from other datasets are added. In the case of the NPD, adding information on the type of school attended (boy's girls, comprehensive, selective, voluntary aided or Academy) is a case in point. Other information might include data on equivalised income in the pupil home neighbourhood, or measures taken from the Index of Multiple Deprivation (IMD) at super output area level. The last two of these are neighbourhood variables, rather than direct measures of pupil characteristics, and they have a value as such. They can also provide a best guess attempt at what individual pupil characteristics might be. There is the obvious risk that our best guess may be wrong, and pages 163 to 190 of DMAG Briefing 2005-31 (Ethnicity and Attainment in Schools) can be read as a worked example, using simple statistics, showing how the assumptions we make about these 'ecological' variables can shape our analyses.

1.6 Creating derived variables for analytical purposes

The SC file extracts released for work in City Hall contains pupil date of birth and date admitted to the current school. From these pupil age when admitted to the current school, and length of time on roll in the current school, can be calculated in SPSS. These are *derived* variables, in the sense that they are derived from other variables in the dataset. They have a particular value in research on pupil mobility. Other derived variables might include measures of entitlement to free school meals over time, or key stage test levels converted into point scores, which can then be arranged in quartiles, quintiles or deciles as circumstances require. Adding data and creating derived variables can go hand in hand. Both are par for the course in work with the NPD, and that is likely to be the case with other post-oven ready datasets.

1.7 Missing data and data quality

One of the advantages of the NPD is that it is a working dataset, in the sense that it is derived from the day-to-day work of education professionals and others in schools. However, as a data warehouse (rather than a single dataset) the NPD is populated by data provided by different people, at different times, and to some extent in different ways. Looked at that way, there is scope for 'discrepancies' in the data, and for missing data. The unsuspecting researcher may well find that datasets which have not been checked for missing values will, at the least, result in totals varying from what is expected. This can be embarrassing if someone else spots that before the research analyst does, and it also risks seriously distorting the analysis. Missing data raise questions about data quality and present issues about how the analysis should proceed. Neither can be taken for granted or ignored.

2. Broader issues

In England the manipulation of abstract symbols in mathematics or in other fields may have higher status than that accorded to detailed understanding of technical procedures. Given that, and what may be humanity's species-specific tendency to see the world in terms of dichotomies or binary oppositions, such as good/bad, raw/cooked, sacred/profane, did/did not obtained 5+ A*-C grades at GCSE, this Guide's focus on 'technical' procedures in SPSS may prompt the assumption that it marginalizes the importance of broader 'professional' research issues. That dichotomy-based inference would be wrong, as dichotomies often are and this Section deals with two (non-dichotomous) issues which can be thrown into particularly sharp relief in work with pupil level data, but which will arise in a range of other settings.

2.1 The means do not justify the ends - data confidentiality

The pupil level data used to provide the worked examples in this Guide are confidential, and that confidentiality has always been maintained at City Hall. However, reports do appear in the national press of confidentiality being breached (elsewhere). There are a variety of reasons why this happens, including what might be called the 'technological fallacy'. This flourishes where too much of the focus is on technical procedures, and where ethical issues simply fall off the radar. At the extreme this can reach the point where agreements entered into are no more than temporary accommodations designed to provide access to otherwise restricted information, or where agreements made in the past by others are disregarded to meet short-term expediency today. There is no reason to assume that this issue is confined to 'technicians' working in technical departments.

Simplified, the technological fallacy has it that if data exist, and if the technology exists to deal with it also exists, then the only issues that arise concern the precise technical means to be used. Once that is decided, whatever analysis follows is OK. A moment's thought should make it clear that it is definitely not OK. The existence of money in banks and of a technology (shotguns) which can be used to separate money from the banks, does not make armed robbery OK. The existence of the means does not justify the ends, and never has.

Some data are confidential and some, but not all, organisations have written and enforceable codes of ethics and standards dealing with this. An enforceable code is one which a junior member of staff can refer to when declining to release confidential information on the instruction of a senior member of staff, and which more senior colleagues would have to accept. The Greater London Authority has such a Code and it is, for example, not only an offence for anyone to release confidential information, it is also an offence to seek access to confidential information without proper authorisation. If your organisation has a code, familiarise yourself with it and abide by it.

Other terms and conditions apply to other datasets. Information from the National Census is available as aggregated Tables and as univariate statistics, with small numbers suppressed to avoid disclosing any individual's identity. It is also available as a sample of anonymised records (SARs), that is as records of specific individuals. As might be expected, terms and conditions apply to ONS data, with SARs being particularly thoroughly policed. If you wish to use census data, find out what those terms and conditions are and observe them.

Commercial datasets, though not bound by exactly the same terms that apply to pupil level data, do have terms of confidentiality attached. PayCheck, which the GLA has access to under contract with a private sector company, provides estimates of income at small area level, and the estimates represent an investment by that company. The investment, and the company's future, would be undercut if one purchaser distributed the data freely to others. As a general principle, contracts should be observed, and one way of securing confidentiality when dealing with others (see the bullet points below) is to write it into a legally binding contract which can be made to stick in a court of law. By contrast, the safest assumption to make about a personal undertaking on confidentiality, given by a member of an organisation which lacks a binding corporate code of ethics, is that it cannot be made to stick.

Some organisations will have computing security procedures in place which prevent access to, or copying of, confidential data. However, if you have a degree of discretion, there are a number of simple points to follow.

- As a general rule, if you are seeking access to data, ask if any terms and conditions apply. If they do, abide by them.
- If you are asked for access to confidential data by someone who does not ask whether terms and conditions apply, tell

them what those terms and conditions are.
- Ensure that you have any terms of confidentiality in writing, that you follow them and that you have copies you can refer others to.
- Do not make unauthorised copies of data, including copies on another computer or on a memory stick for work elsewhere.
- Never pass confidential data to others without full clearance.
- Never pass sensitive data to an organisation which lacks an enforceable corporate Code of Ethics. Departmental codes of ethics, or someone's word of honour, are no substitute for an enforceable corporate code.
- Where terms and conditions apply to the circulation of analyses of data, observe them.
- Work in a secure environment where confidential data are not accessible to others – do not, for example, leave computing discs lying around, or leave data unattended in plain sight on the screen of an unlocked computer.
- Password protect confidential files, but only if you have the authorisation to do so.

2.2 Understanding the data

Organisations exist which are large enough to support a marked division of labour. Large or highly specialised social research organisations may, for example, have; teams of interviewers, data coders, analysts, supervisors and managers/team leaders who organise contact with the outside world. Some, and perhaps the majority of, research analysts will not be in that situation. In those instances, and regardless of professional seniority, individuals may find that they need to be able to work across a range of activities, including the readying of data for analysis. That work requires an understanding of the data being handled. This is essential, not optional. Additionally, research analysts will need to organise themselves if the are to acquire that understanding, and the way dataset user manuals are approached illustrates some of the issues at stake.

Data on occupation in the 2001 national census provides one example. A revised occupational scheme was arrived at after considerable discussion, some of which is set out in *The ESRC Review of Government Social Classifications*. That review also contains references to other published material relevant to the issue: the review is available at the time of writing at http://www.statistics.gov.uk/statbase/Product.asp?vlnk=2416&More=N.

The final report from ESRC (with a user manual) is available at the time of writing at http://www.statistics.gov.uk/StatBase/Product.asp?vlnk=14066

There is no NPD user manual, and there is more than one view on whether that is a good or a bad thing. One view, which differs from but does not necessarily contradict the views in the Guide, is given in Hansen and Vignoles 'The use of large-scale data sets in educational research.' London TLRP, 2007. This is available at the time of writing at www.bera.ac.uk /the-use-of-large-scale-data-sets-in-educational-research/. I do not assume that the authors are open to the criticisms I make later, but nonetheless their report on the BERA website contains the following:

Another need is for better dissemination of information on data availability …..Whilst guidance notes for particular surveys, such as the National Child Development Study, are generally comprehensive, information on particular fields in administrative datasets, such as the Pupil Level Annual Schools Census (PLASC)/National Pupil Database (NPD), is often weak

If this view advocates dialogue, then all to the best, and the view taken here is that research analysts do indeed need to be outward looking and talk to others. However, and contrary to what we might conclude from the quote above, there is already a great deal of information on the areas (variables/fields) covered by the NPD, and it is freely available for research analysts to read. That material is available in part on the Internet at sites listed in DMAG 2005 – 8 (though the reader may now need to update some of the references). Put another way, the information research analysts need is there, it is just not in a single codified user manual. That being so, the issue is *not* whether information available to research analysts needs to be strengthened, but rather whether it should be brought together in a single document, and it is far from self-evident that one codified document would be superior to what already exists or that it would be easy to produce.

Essentially, the view taken here is that a user manual for the NPD in particular *may* help those research analysts who have *already* shown initiative in getting to grips with the data, using the professional sources that are already available. A detailed manual will simply be one more string to their bow. If, on the other hand, the belief is that there is a single norm for datasets (and which can be summed up in the question, 'where's the user manual?'), then four problems can arise.

The first problem concerns acquiring and developing an understanding of data in datasets such as the NPD. I doubt whether those who

wrote the user guides to NS-SEC or to NCDS would claim that these contain 'everything'. Those wishing to work in a particular research field will need to keep up with the research literature, and research analysts would take that as read. Further, that is something individuals do for themselves, rather than something others do for them. Similarly, those wishing to work can usefully keep up with the equivalent of the information provided on the websites listed in DMAG Briefing 2005 – 08.

Indeed, there are real advantages in the present situation, where advice on the data drawn on by the NPD is available in publications for different groups of education specialists. Section 6 of DMAG Briefing 2005 – 08 makes a series of points which are relevant at this stage.

…the data flows which feed in to the NPD tend to predate it, were designed for purposes other than populating the NPD, and … the provision of data is a spin-off from the work of those involved, rather than being their main function. Guidance for a range of education specialists, which ultimately leads to data being made available for the NPD, is set in the context of the main, educational, responsibilities of those specialists. Research analysts who are familiar with that guidance will have a deeper understanding of the education issues involved than will those who do not.

We might modify the end of that last sentence to read 'will have an understanding of the data in the NPD which others will lack'.

There will also be instances where researchers will need the co-operation of others, if only to gain understanding of the datasets that are being worked with. As far as the NPD is concerned, the single largest group of experts on the data will be teachers, the educational specialists referred to above, followed by local authority education research and statistics staff. The single smallest group is in regional government.

Those with a purely university background might usefully consider how members of either of the first two groups are likely to react when the pupil level information they use in professional work is described as 'administrative data', particularly when those individuals are aware that university researchers tend to work with comparatively small datasets, often from sample surveys which are difficult to update if they are updated at all, and which provide no sure guide to the situation in individual classrooms. (Also see Section 1.1 above).

.

There is a serious risk that research analysts who do not fully understand the data they are dealing with will simply come unstuck. An individual who hopes to get by on the basis of the limited understanding that may come through reading a user data manual, is in as much of a pickle as someone who assumes that following a recipe from Elizabeth David's *French Provincial Cooking* (Penguin, Harmondsworth, 1970) means that he or she will be transformed into a great chef, with no need to understand how best to judge the quality of produce, or what produce from where is used when, or how it should be prepared for cooking, or what the difference is between the French and English approach to jointing meat. That individual may (but only may) be able to produce a passable version of a particular dish on this side of the Channel, but will never be a great chef (or cut much of a dash as a cook on the other side of the Channel) and may unintentionally prompt hilarity at his or her own expense.

Secondly, if researchers' a person's experience and horizons are limited to oven ready datasets, and they cannot imagine possibilities beyond that horizon, then they will face real difficulties in coming to terms with the world in which datasets such as the NPD exist. Where the aim is to foster the skills and initiative needed by individuals to move on from university oven-ready datasets (with their child's bicycle training wheels on) to real world datasets (with the training wheels off) then a user manual would be a liability if it perpetuated rather than corrected the assumption that a user manual is 'the norm' and/or that the provision of detailed documentation was always and everywhere somebody else's responsibility.

The point made in Section 1.1 still applies. Skills that are needed as a matter of course in work with pupil level data turned out to be in short supply at a point in time when they were needed. Where individuals in or leaving university also expect data user guides to be there as a matter of course, and cannot imagine life beyond them, then that skills gap widens to a possibly unbridgeable extent. Creating a myth that user data manuals are the norm has its drawbacks.

In setting out the first two problems that can arise, I have deliberately used language which aims to shock complacency about the place of data user guides off the stage, and to try and provide a different perspective on the standing of datasets such as the NPD. The key message is that the NPD has major strengths, and these should not be overlooked and/or neglected in pursuit of a user data manual which will not provide guidance up to the standard achieved in the professional documentation. A data guide which provides an audit trail showing where the data come from and, if they are derived data, how they have been calculated, is another matter.

The third problem is more prosaic, but will be recognised by those who have worked with longitudinal data. The issue is one of time, though on this occasion it is research analyst time rather than the time of those on whose work the NPD relies. The NPD is now updated three times a year. Data definitions and the variables included can and do change over time, and research analysts working on pupil level datasets from different years will need to keep pace with developments. In any busy research group there will be little, if any scope, for individuals to take extensive periods of 'time out' for training in what the data actually mean and what their context is. The development of an understanding of what the data are about is ongoing, and is manageable by researchers themselves on that basis.

Lastly, there may in any event be difficulties in arriving at a single NPD national data user guide within a reasonable time span. The NPD is not a single simple dataset, produced by one agency working on a narrow range of data and to one timetable. Sections 5, 6, 7, 8 and 9 in DMAG Briefing 2005 – 8 stress that a number of agencies are involved in work on which the NPD draws. Each of these will be working to their own timetables and imperatives, and issues around key stage 3 assessments in 2008 provide a reminder of how strong those imperatives can be. The annual brouhaha as pupils receive their public examination results further illustrates how important those timetables can be to pupils and parents (and university admissions tutors). The agencies involved will *of necessity* have to put their own main responsibilities first, and we cannot realistically expect them to reorder those priorities to co-ordinate and produce a single guide to the existing guides. Synchronising these will be more difficult than a brief encounter with the data in a seminar presentation might suggest.

The unavoidable point is that organising data and readying it for analysis presupposes that research analysts have the energy and imagination needed to organise themselves. This will involve a degree of initiative on somebody's part in actively seeking out information on the meaning and provenance of data in datasets. It also means that the individual researcher, if working alone, or someone within a team if a group effort is involved, will have done their homework on the provenance of data and can build rather than burn bridges with those who are experts on the data and can provide advice. However managed, organising data is an activity embedded in a range of qualitatively different activities and considerations: it is not part of a subset of narrowly 'technical' exercises.

Given the points made in this Section, it should come as no surprise that this threshold Guide is not a NPD user manual and that I do not plan to write one. For those just starting work with the NPD, DMAG Briefing 2005 – 08, mentioned above, does provide a broad (threshold) introduction, and gives one way in to a complex body of data. At the time of writing, that and other DMAG Briefings referred to in the Guide are available on request. Contact details are given on the Guide's inside front page.

3. The world after oven ready datasets.

Section 1 stressed, though somewhat in the abstract, that some datasets are held in several files within relational databases, and that relational databases are more likely than SPSS to make use of text (string) data. It also stressed that at least some variables from relational databases will need to be labelled if they are to become meaningful to the user, and that the potential of secondary datasets may be enhanced by the inclusion of derived variables and variables from other datasets.

The National Pupil Dataset is held in a relational data warehouse, and Figure 1 illustrates the point that data released for analysis are held in multiple files. With their .txt suffix, it also illustrates that files are not released in SPSS format.

So why 'disperse' information between so many files? To provide a real world, rather than a wholly abstract view, Figure 1 lists the files in a 2006 NPD extract released by DCSF. Each row in the *KS4Res_2006_London.txt* file is a record of an individual public examination subject taken by pupils in the maintained sector, and contains short text codes of the individual subject. Where a pupil has ten examination entries, there are ten rows showing the individual subjects entered. Keeping full subject names in *KS4Res_2006_London.txt* will increase demands on computing capacity from what is already a very large file, and full subject names, with the short subject code, is kept in the separate *MappingCodes.txt* file. The short code acts as a link variable between *KS4Res_2006_London.txt* and *MappingCodes.txt,* and the long subject names are only called up as and when a tabulation of performance in individual subjects is required. This eases demands on computing capacity, and this way of organising examination data predates the NPD by well over ten years. It follows the format used in files used by the locally authority-funded National Consortium for Examination Results. This is another example of a dataset which was not in SPSS, would not be called a research dataset, and yet had considerable value for research.

SPSS, as noted, has 'built in' statistical programs that relational databases lack. It also, again as noted, takes a different approach from relational databases in that analyses are carried out on data in a single flat files.

Figure 2 shows the variables released from the core 2007 'PLASC' file. The information shown in the 'Label' part of the Figure has been added in City Hall to make the variables more meaningful to the reader, and is not present in the file extract received from DCSF. (It is present in an EXCEL variable list provided separately by DCSF). You will note that the file as received has no value labels.

There are only 63 variables in the 'PLASC' file. However, even for the education specialists, some of the variables names shown in Figure 2 will not be particularly meaningful on their own (or after six months) What information do the variables cti_07, sdte_07 or trcg_07 contain? One option is to change the names to something more meaningful so that we do not find ourselves perpetually scrolling through the variables in SPSS Variable View while consulting the DCSF EXCEL variable list. However, there is a limit to how long those variable names can be, and while longer names may help the individual researcher, they may well mean nothing to the people who read the output he or she produces.

This is the point at which someone adds the information in the SPSS 'Labels' area. That information will appear in SPSS output, and will add meaning for others. When lists of variables are shown in SPSS procedures, the information in the 'Labels' section will be shown instead of the variable name, and but there will be precious little space in dialogue boxes to show that name. Choosing variable labels carefully so that they can be identified in dialogue boxes is something of an art, and worth cultivating.

That said it ought to be clear, even from the abbreviated variable names in Figure 2, that the 'PLASC' file contains neither key stage assessment nor public examination records. As we might now expect, and as Figure 1 indicates, these records are kept in separate files, and that information will, somehow, need to be added to the 'PLASC' file.

Figure 3 shows one instance of a value label that has been added to the variable, in this case to the variable containing information on whether or not a pupil obtained 5 or more higher grade passes at GCSE (public examinations) including English and mathematics. The code 0 now has the label 'Pupil did not achieve these passes' and the code 1 has the value label 'Pupil did achieve these passes'. This information has been added once files are received from DCSF, and will be shown in output. Without these value labels, output would be in the form of '0' and '1', which would not be meaningful to the reader. This example also provides a reminder that string codes in DCSF (or other) files are best replaced by numeric equivalents (with value labels added).

Additionally, a range of information that we might take for granted would be included in an education dataset is neither listed in Figure 2, and it also not present in the variables shown in

Figure 1. In a list that is not exhaustive, the 'missing' information includes

- name of the school attended
- name of the school local authority
- type of school attended – primary, secondary or special
- type of school attended, community, voluntary aided, foundation, Academy, CTC
- type of school attended, denomination
- type of school attend, specialism
- type of school attended, gender of intake
- school key stage and public examination 'league table' figures
- pupil home ward
- pupil home local authority area
- pupil age when first admitted to the current school
- pupil length of time on roll in the current school
- distance between pupil home and school attended
- pupil home neighbourhood characteristics

Figure 4 shows that some of these variables have been created. They have somehow been 'found' outside the NPD and added to the NPD file extracts, or have been created as derived variables or both. The name of the datasets shown in Figures 3 and 4 at the top of the two Figures is '*new merged trimmed 2002 to 2005 LPD version 2A*'. This is a longitudinal dataset. The records from four separate years have been merged together (albeit in a trimmed form to reduce demands on computing capacity). Researchers should also take it as read that there would need to be checks in place as work proceeds on the quality and completeness of data.

The list of variables in that dataset is shown in the Appendix, partly to show the range of information involved, but also to highlight the distance that has been travelled from the situation illustrated in Figure 2. (How that list is obtained is shown in Figure 5. Don't worry if the Figure does not mean much at this stage. It will, before the end of the Guide is reached).

To summarise, working with secondary datasets, and work with secondary datasets drawn from relational datasets is likely to involve

1. importing files in other formats into SPSS, without losing information along the way
2. adding variables from other files to the core dataset where data are held in multiple files in a relational database
3. locating other datasets containing variables that will add value to the core dataset, and merging relevant variables with the core dataset
4. merging the record from more than one year to create a longitudinal dataset
5. adding variable labels to make variable names meaningful
6. converting string data into numeric data
7. adding value labels to make output meaningful
8. creating derived variables
9. checking the completeness and quality of the record
10. working with computing constraints in mind.

The Guide introduces each of these on a step by step basis, using worked examples. The pupil level data used to provide those worked examples in the Guide are based on anonymised records of each individual child in the maintained school system in England, and some data are sensitive. Unsurprisingly, pupil level data are not available on demand.

Access to pupil level data is by application to DCSF, and a successful application has been made each year since 2002 at City Hall. This has included applications for access to variables which are sensitive and not normally released to researchers. This means that *before* the researcher gets to the point of readying data, let alone analysing it and providing reports, he or she must be able to make a satisfactory application, and be able to meet stringent standards in respect of particularly sensitive data.

The point was stressed in Sections 1 and 2, and is made again here. Readying data for analysis is not a self-contained activity, which can be carried out in isolation by an otherwise unaware technician. It is located within the wider research cycle, and needs to be seen in those terms.

Figure 1. 2006 NPD data files

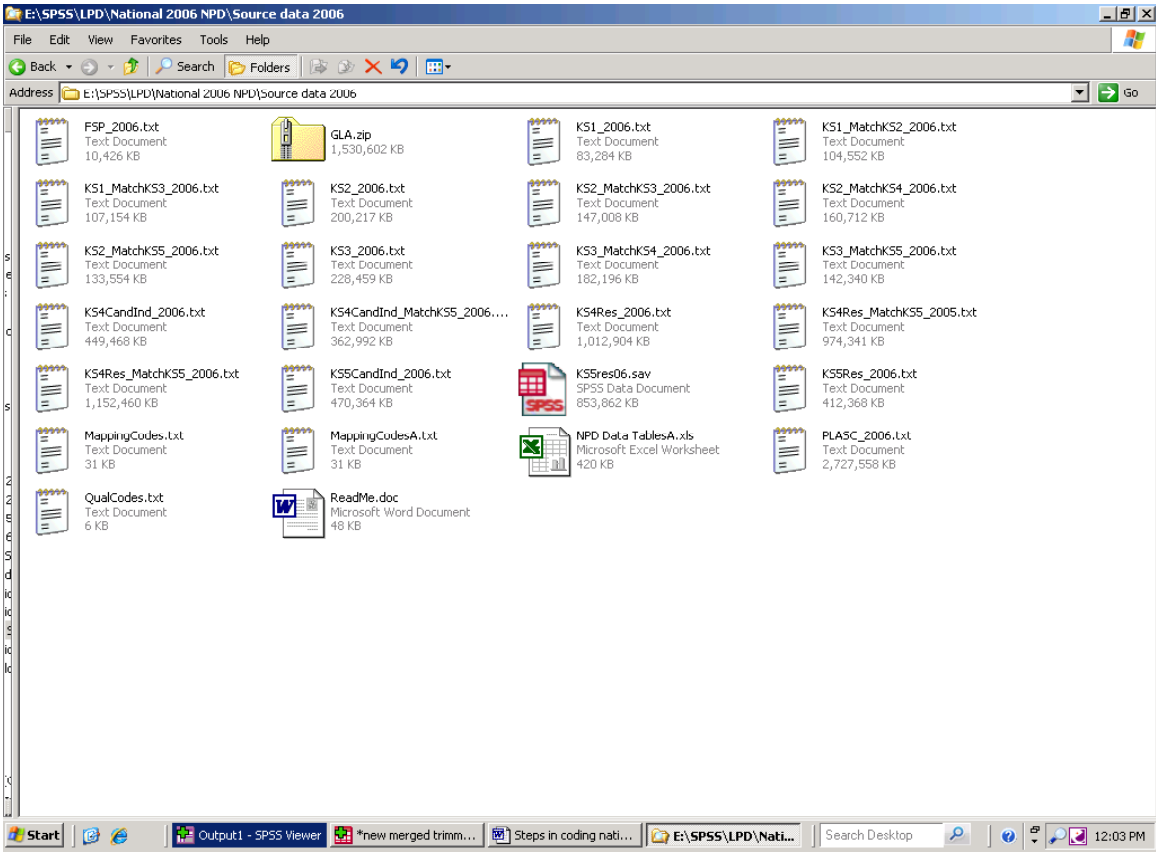
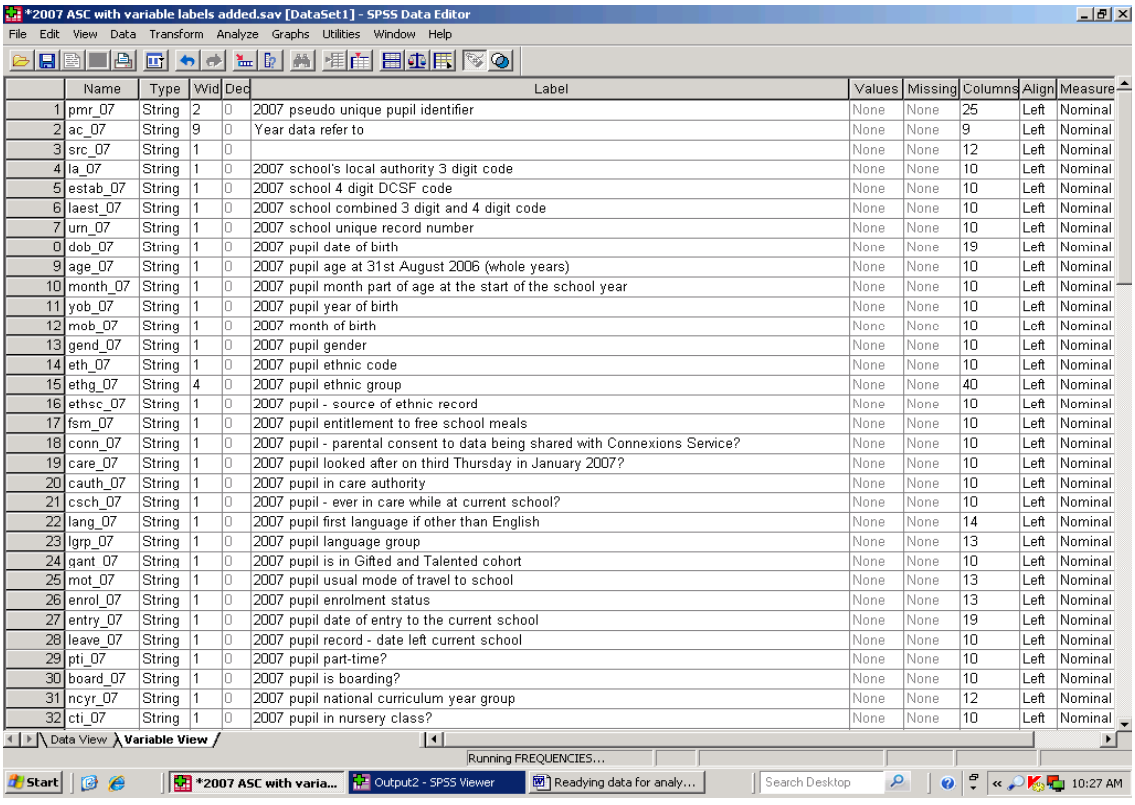
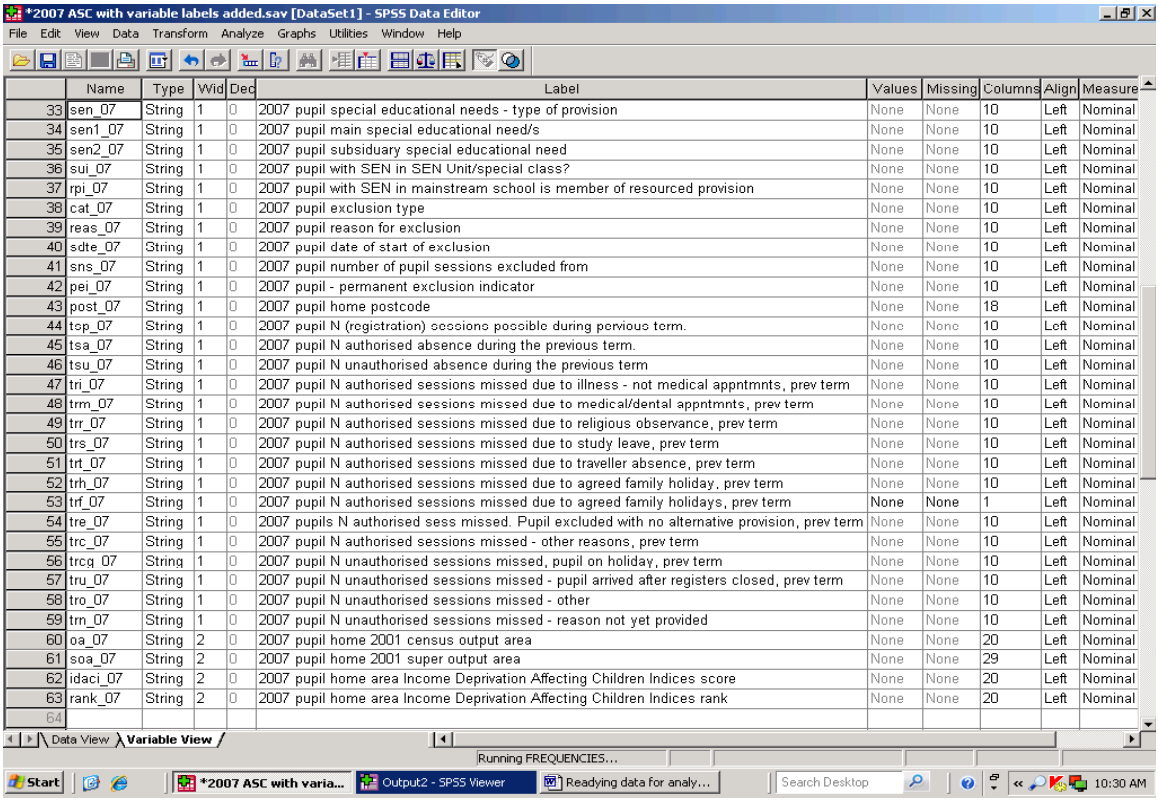


Figure 2. 'Before' - The Variable View of the 2007 DCSF "PLASC" extract. (Variable labels are not included in DCSF extract).



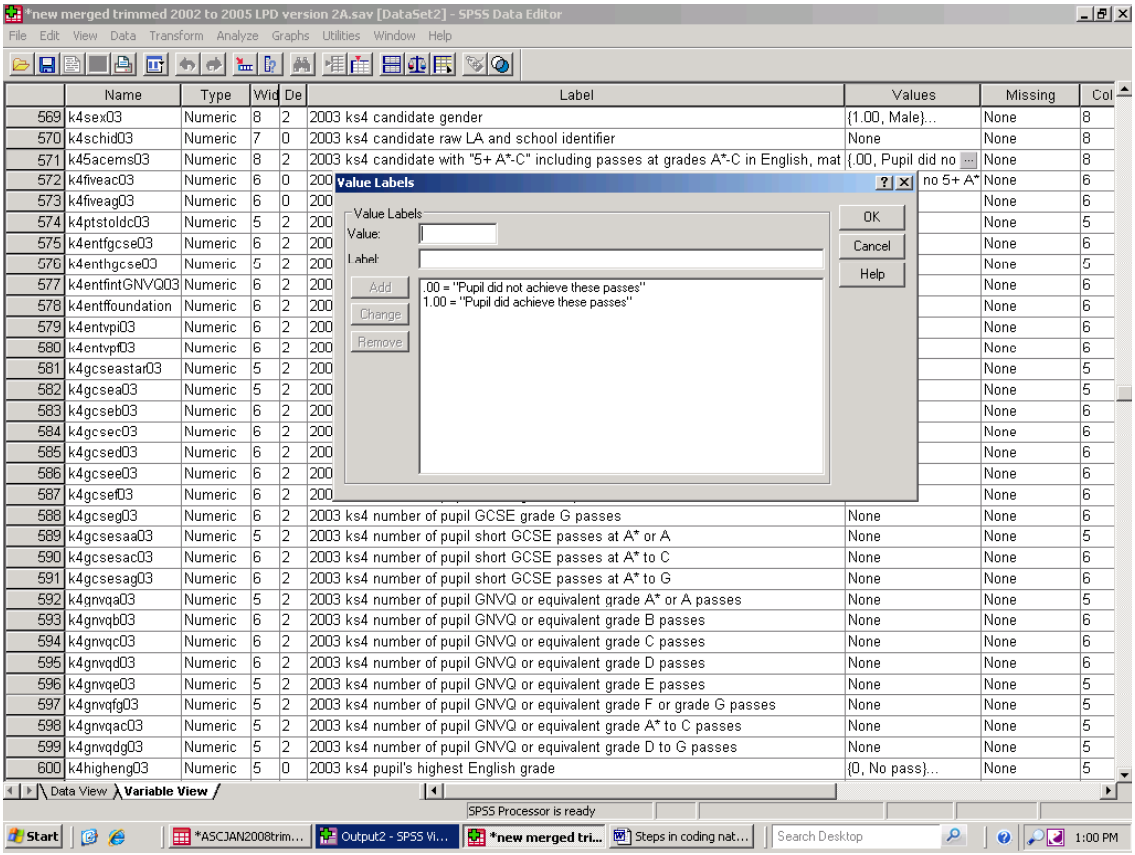
	Name	Type	Wid	Dec	Label	Values	Missing	Columns	Align	Measure
1	pmr_07	String	2	0	2007 pseudo unique pupil identifier	None	None	25	Left	Nominal
2	ac_07	String	9	0	Year data refer to	None	None	9	Left	Nominal
3	src_07	String	1	0		None	None	12	Left	Nominal
4	la_07	String	1	0	2007 school's local authority 3 digit code	None	None	10	Left	Nominal
5	estab_07	String	1	0	2007 school 4 digit DCSF code	None	None	10	Left	Nominal
6	laest_07	String	1	0	2007 school combined 3 digit and 4 digit code	None	None	10	Left	Nominal
7	urn_07	String	1	0	2007 school unique record number	None	None	10	Left	Nominal
0	dob_07	String	1	0	2007 pupil date of birth	None	None	19	Left	Nominal
9	age_07	String	1	0	2007 pupil age at 31st August 2006 (whole years)	None	None	10	Left	Nominal
10	month_07	String	1	0	2007 pupil month part of age at the start of the school year	None	None	10	Left	Nominal
11	yob_07	String	1	0	2007 pupil year of birth	None	None	10	Left	Nominal
12	mob_07	String	1	0	2007 month of birth	None	None	10	Left	Nominal
13	gend_07	String	1	0	2007 pupil gender	None	None	10	Left	Nominal
14	eth_07	String	1	0	2007 pupil ethnic code	None	None	10	Left	Nominal
15	ethg_07	String	4	0	2007 pupil ethnic group	None	None	40	Left	Nominal
16	ethsc_07	String	1	0	2007 pupil - source of ethnic record	None	None	10	Left	Nominal
17	fsm_07	String	1	0	2007 pupil entitlement to free school meals	None	None	10	Left	Nominal
18	conn_07	String	1	0	2007 pupil - parental consent to data being shared with Connexions Service?	None	None	10	Left	Nominal
19	care_07	String	1	0	2007 pupil looked after on third Thursday in January 2007?	None	None	10	Left	Nominal
20	cauth_07	String	1	0	2007 pupil in care authority	None	None	10	Left	Nominal
21	csch_07	String	1	0	2007 pupil - ever in care while at current school?	None	None	10	Left	Nominal
22	lang_07	String	1	0	2007 pupil first language if other than English	None	None	14	Left	Nominal
23	lgrp_07	String	1	0	2007 pupil language group	None	None	13	Left	Nominal
24	gant_07	String	1	0	2007 pupil is in Gifted and Talented cohort	None	None	10	Left	Nominal
25	mot_07	String	1	0	2007 pupil usual mode of travel to school	None	None	13	Left	Nominal
26	enrol_07	String	1	0	2007 pupil enrolment status	None	None	13	Left	Nominal
27	entry_07	String	1	0	2007 pupil date of entry to the current school	None	None	19	Left	Nominal
28	leave_07	String	1	0	2007 pupil record - date left current school	None	None	10	Left	Nominal
29	pti_07	String	1	0	2007 pupil part-time?	None	None	10	Left	Nominal
30	board_07	String	1	0	2007 pupil is boarding?	None	None	10	Left	Nominal
31	ncyr_07	String	1	0	2007 pupil national curriculum year group	None	None	12	Left	Nominal
32	cti_07	String	1	0	2007 pupil in nursery class?	None	None	10	Left	Nominal

Figure 2. 'Before' - The Variable View of the 2007 DCSF "PLASC" extract, continued. (Variable labels are not included in the DCSF extract)



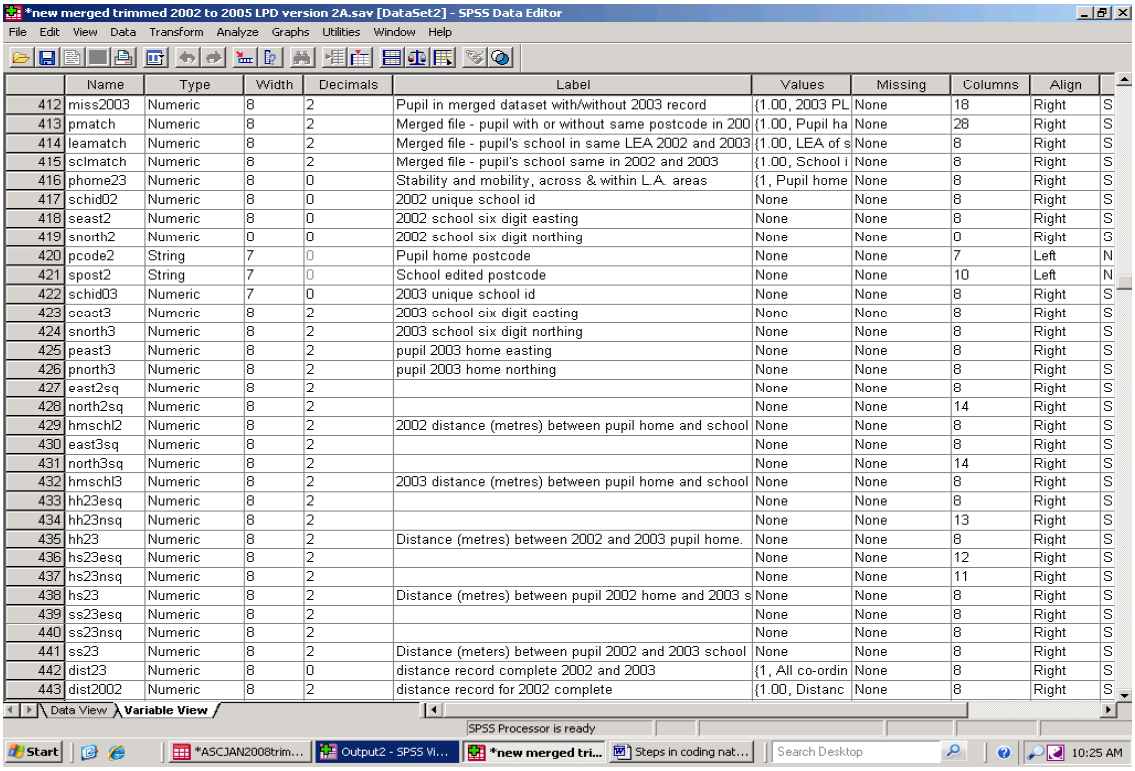
	Name	Type	Wid	Dec	Label	Values	Missing	Columns	Align	Measure
33	sen_07	String	1	0	2007 pupil special educational needs - type of provision	None	None	10	Left	Nominal
34	sen1_07	String	1	0	2007 pupil main special educational need/s	None	None	10	Left	Nominal
35	sen2_07	String	1	0	2007 pupil subsiduary special educational need	None	None	10	Left	Nominal
36	sui_07	String	1	0	2007 pupil with SEN in SEN Unit/special class?	None	None	10	Left	Nominal
37	rpi_07	String	1	0	2007 pupil with SEN in mainstream school is member of resourced provision	None	None	10	Left	Nominal
38	cat_07	String	1	0	2007 pupil exclusion type	None	None	10	Left	Nominal
39	reas_07	String	1	0	2007 pupil reason for exclusion	None	None	10	Left	Nominal
40	sdte_07	String	1	0	2007 pupil date of start of exclusion	None	None	10	Left	Nominal
41	sns_07	String	1	0	2007 pupil number of pupil sessions excluded from	None	None	10	Left	Nominal
42	pei_07	String	1	0	2007 pupil - permanent exclusion indicator	None	None	10	Left	Nominal
43	post_07	String	1	0	2007 pupil home postcode	None	None	18	Left	Nominal
44	tsp_07	String	1	0	2007 pupil N (registration) sessions possible during pervious term.	None	None	10	Left	Nominal
45	tsa_07	String	1	0	2007 pupil N authorised absence during the previous term.	None	None	10	Left	Nominal
46	tsu_07	String	1	0	2007 pupil N unauthorised absence during the previous term	None	None	10	Left	Nominal
47	tri_07	String	1	0	2007 pupil N authorised sessions missed due to illness - not medical appntmnts, prev term	None	None	10	Left	Nominal
48	trm_07	String	1	0	2007 pupil N authorised sessions missed due to medical/dental appntmnts, prev term	None	None	10	Left	Nominal
49	trr_07	String	1	0	2007 pupil N authorised sessions missed due to religious observance, prev term	None	None	10	Left	Nominal
50	trs_07	String	1	0	2007 pupil N authorised sessions missed due to study leave, prev term	None	None	10	Left	Nominal
51	trt_07	String	1	0	2007 pupil N authorised sessions missed due to traveller absence, prev term	None	None	10	Left	Nominal
52	trh_07	String	1	0	2007 pupil N authorised sessions missed due to agreed family holiday, prev term	None	None	10	Left	Nominal
53	trf_07	String	1	0	2007 pupil N authorised sessions missed due to agreed family holidays, prev term	None	None	1	Left	Nominal
54	tre_07	String	1	0	2007 pupils N authorised sess missed. Pupil excluded with no alternative provision, prev term	None	None	10	Left	Nominal
55	trc_07	String	1	0	2007 pupil N authorised sessions missed - other reasons, prev term	None	None	10	Left	Nominal
56	trcg_07	String	1	0	2007 pupil N unauthorised sessions missed, pupil on holiday, prev term	None	None	10	Left	Nominal
57	tru_07	String	1	0	2007 pupil N unauthorised sessions missed - pupil arrived after registers closed, prev term	None	None	10	Left	Nominal
58	tro_07	String	1	0	2007 pupil N unauthorised sessions missed - other	None	None	10	Left	Nominal
59	trn_07	String	1	0	2007 pupil N unauthorised sessions missed - reason not yet provided	None	None	10	Left	Nominal
60	oa_07	String	2	0	2007 pupil home 2001 census output area	None	None	20	Left	Nominal
61	soa_07	String	2	0	2007 pupil home 2001 super output area	None	None	29	Left	Nominal
62	idaci_07	String	2	0	2007 pupil home area Income Deprivation Affecting Children Indices score	None	None	20	Left	Nominal
63	rank_07	String	2	0	2007 pupil home area Income Deprivation Affecting Children Indices rank	None	None	20	Left	Nominal
64										

Figure 3. 'After'. 2003 Merged 2002 to 2005 LPD with selected 2003 key stage 4 variables



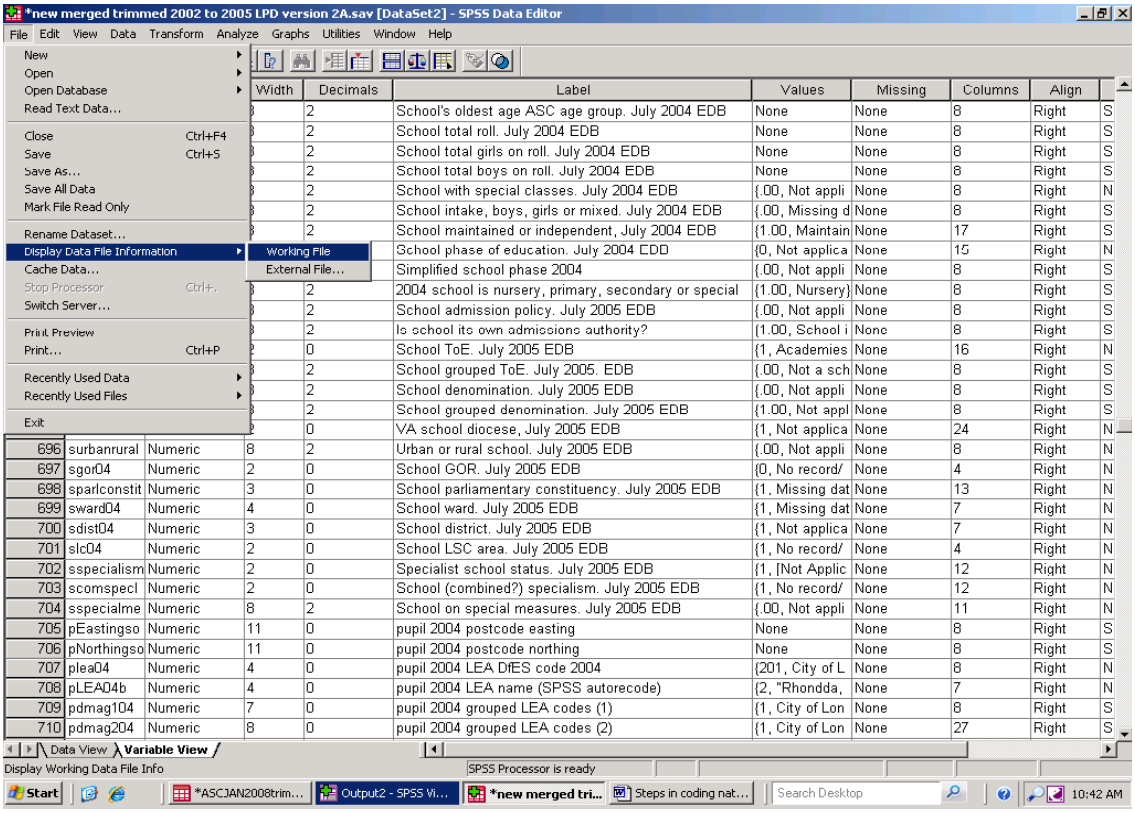
	Name	Type	Wid	De	Label	Values	Missing	Col
569	k4sex03	Numeric	8	2	2003 ks4 candidate gender	{1.00, Male}...	None	8
570	k4schid03	Numeric	7	0	2003 ks4 candidate raw LA and school identifier	None	None	8
571	k45acems03	Numeric	8	2	2003 ks4 candidate with "5+ A*-C" including passes at grades A*-C in English, mat	{.00, Pupil did no	None	8
572	k4fiveac03	Numeric	6	0	200	no 5+ A*	None	6
573	k4fiveag03	Numeric	6	0	200		None	6
574	k4ptstoldc03	Numeric	5	2	200		None	5
575	k4entfgcse03	Numeric	6	2	200		None	6
576	k4enthgcse03	Numeric	5	2	200		None	5
577	k4entfintGNVQ03	Numeric	6	2	200		None	6
578	k4entffoundation	Numeric	6	2	200		None	6
579	k4entvpi03	Numeric	6	2	200		None	6
580	k4entvpf03	Numeric	6	2	200		None	6
581	k4gcseastar03	Numeric	5	2	200		None	5
582	k4gcsea03	Numeric	5	2	200		None	5
583	k4gcseb03	Numeric	6	2	200		None	6
584	k4gcsec03	Numeric	6	2	200		None	6
585	k4gcsed03	Numeric	6	2	200		None	6
586	k4gcsee03	Numeric	6	2	200		None	6
587	k4gcsef03	Numeric	6	2	200		None	6
588	k4gcseg03	Numeric	6	2	2003 ks4 number of pupil GCSE grade G passes	None	None	6
589	k4gcsesaa03	Numeric	5	2	2003 ks4 number of pupil short GCSE passes at A* or A	None	None	5
590	k4gcsesac03	Numeric	6	2	2003 ks4 number of pupil short GCSE passes at A* to C	None	None	6
591	k4gcsesag03	Numeric	6	2	2003 ks4 number of pupil short GCSE passes at A* to G	None	None	6
592	k4gnvqa03	Numeric	5	2	2003 ks4 number of pupil GNVQ or equivalent grade A* or A passes	None	None	5
593	k4gnvqb03	Numeric	6	2	2003 ks4 number of pupil GNVQ or equivalent grade B passes	None	None	6
594	k4gnvqc03	Numeric	6	2	2003 ks4 number of pupil GNVQ or equivalent grade C passes	None	None	6
595	k4gnvqd03	Numeric	6	2	2003 ks4 number of pupil GNVQ or equivalent grade D passes	None	None	6
596	k4gnvqe03	Numeric	5	2	2003 ks4 number of pupil GNVQ or equivalent grade E passes	None	None	5
597	k4gnvqfg03	Numeric	5	2	2003 ks4 number of pupil GNVQ or equivalent grade F or grade G passes	None	None	5
598	k4gnvqac03	Numeric	5	2	2003 ks4 number of pupil GNVQ or equivalent grade A* to C passes	None	None	5
599	k4gnvqdg03	Numeric	5	2	2003 ks4 number of pupil GNVQ or equivalent grade D to G passes	None	None	5
600	k4highengl03	Numeric	5	0	2003 ks4 pupil's highest English grade	{0, No pass}...	None	5

Figure 4. More 'after' - Selected variables in the Merged 2002 to 2005 LPD



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	
412	miss2003	Numeric	8	2	Pupil in merged dataset with/without 2003 record	{1.00, 2003 PL	None	18	Right	S
413	pmatch	Numeric	8	2	Merged file - pupil with or without same postcode in 200	{1.00, Pupil ha	None	28	Right	S
414	leamatch	Numeric	8	2	Merged file - pupil's school in same LEA 2002 and 2003	{1.00, LEA of s	None	8	Right	S
415	sclmatch	Numeric	8	2	Merged file - pupil's school same in 2002 and 2003	{1.00, School i	None	8	Right	S
416	phome23	Numeric	8	0	Stability and mobility, across & within L.A. areas	{1, Pupil home	None	8	Right	S
417	schid02	Numeric	8	0	2002 unique school id	None	None	8	Right	S
418	seast2	Numeric	8	0	2002 school six digit easting	None	None	8	Right	S
419	snorth2	Numeric	0	0	2002 school six digit northing	None	None	0	Right	S
420	pcode2	String	7	0	Pupil home postcode	None	None	7	Left	N
421	spost2	String	7	0	School edited postcode	None	None	10	Left	N
422	schid03	Numeric	7	0	2003 unique school id	None	None	8	Right	S
423	seast3	Numeric	8	2	2003 school six digit easting	None	None	8	Right	S
424	snorth3	Numeric	8	2	2003 school six digit northing	None	None	8	Right	S
425	peast3	Numeric	8	2	pupil 2003 home easting	None	None	8	Right	S
426	pnorth3	Numeric	8	2	pupil 2003 home northing	None	None	8	Right	S
427	east2sq	Numeric	8	2		None	None	8	Right	S
428	north2sq	Numeric	8	2		None	None	14	Right	S
429	hmschl2	Numeric	8	2	2002 distance (metres) between pupil home and school	None	None	8	Right	S
430	east3sq	Numeric	8	2		None	None	8	Right	S
431	north3sq	Numeric	8	2		None	None	14	Right	S
432	hmschl3	Numeric	8	2	2003 distance (metres) between pupil home and school	None	None	8	Right	S
433	hh23esq	Numeric	8	2		None	None	8	Right	S
434	hh23nsq	Numeric	8	2		None	None	13	Right	S
435	hh23	Numeric	8	2	Distance (metres) between 2002 and 2003 pupil home.	None	None	8	Right	S
436	hs23esq	Numeric	8	2		None	None	12	Right	S
437	hs23nsq	Numeric	8	2		None	None	11	Right	S
438	hs23	Numeric	8	2	Distance (metres) between pupil 2002 home and 2003 s	None	None	8	Right	S
439	ss23esq	Numeric	8	2		None	None	8	Right	S
440	ss23nsq	Numeric	8	2		None	None	8	Right	S
441	ss23	Numeric	8	2	Distance (meters) between pupil 2002 and 2003 school	None	None	8	Right	S
442	dist23	Numeric	8	0	distance record complete 2002 and 2003	{1, All co-ordin	None	8	Right	S
443	dist2002	Numeric	8	2	distance record for 2002 complete	{1.00, Distanc	None	8	Right	S

Figure 5. Obtaining a list of variables – the Merged 2002 to 2005 LPD



	Width	Decimals	Label	Values	Missing	Columns	Align	
	8	2	School's oldest age ASC age group. July 2004 EDB	None	None	8	Right	S
	8	2	School total roll. July 2004 EDB	None	None	8	Right	S
	8	2	School total girls on roll. July 2004 EDB	None	None	8	Right	S
	8	2	School total boys on roll. July 2004 EDB	None	None	8	Right	S
	8	2	School with special classes. July 2004 EDB	{.00, Not appli	None	8	Right	N
	8	2	School intake, boys, girls or mixed. July 2004 EDB	{.00, Missing d	None	8	Right	S
	8	2	School maintained or independent, July 2004 EDB	{1.00, Maintain	None	17	Right	S
	8	2	School phase of education. July 2004 EDD	{0, Not applica	None	15	Right	N
	8	2	Simplified school phase 2004	{.00, Not appli	None	8	Right	S
	8	2	2004 school is nursery, primary, secondary or special	{1.00, Nursery}	None	8	Right	S
	8	2	School admission policy. July 2005 EDB	{.00, Not appli	None	8	Right	S
	8	2	Is school its own admissions authority?	{1.00, School i	None	8	Right	S
	8	0	School ToE. July 2005 EDB	{1, Academies	None	16	Right	N
	8	2	School grouped ToE. July 2005. EDB	{.00, Not a sch	None	8	Right	S
	8	2	School denomination. July 2005 EDB	{.00, Not appli	None	8	Right	S
	8	2	School grouped denomination. July 2005 EDB	{1.00, Not appl	None	8	Right	S
	8	0	VA school diocese, July 2005 EDB	{1, Not applica	None	24	Right	N
surbanrural	8	2	Urban or rural school. July 2005 EDB	{.00, Not appli	None	8	Right	N
sgor04	2	0	School GOR. July 2005 EDB	{0, No record/	None	4	Right	N
sparlconstit	3	0	School parliamentary constituency. July 2005 EDB	{1, Missing dat	None	13	Right	N
sward04	4	0	School ward. July 2005 EDB	{1, Missing dat	None	7	Right	N
sdist04	3	0	School district. July 2005 EDB	{1, Not applica	None	7	Right	N
slc04	2	0	School LSC area. July 2005 EDB	{1, No record/	None	4	Right	N
sspecialism	2	0	Specialist school status. July 2005 EDB	{1, [Not Applic	None	12	Right	N
scomspecl	2	0	School (combined?) specialism. July 2005 EDB	{1, No record/	None	12	Right	N
sspecialme	8	2	School on special measures. July 2005 EDB	{.00, Not appli	None	11	Right	N
pEastingso	11	0	pupil 2004 postcode easting	None	None	8	Right	S
pNorthingso	11	0	pupil 2004 postcode northing	None	None	8	Right	S
plea04	4	0	pupil 2004 LEA DfES code 2004	{201, City of L	None	8	Right	N
pLEA04b	4	0	pupil 2004 LEA name (SPSS autorecode)	{2, "Rhondda,	None	7	Right	N
pdmag104	7	0	pupil 2004 grouped LEA codes (1)	{1, City of Lon	None	8	Right	S
pdmag204	8	0	pupil 2004 grouped LEA codes (2)	{1, City of Lon	None	27	Right	S

4. Taking text (.txt) files into SPSS – File/Read text data

The files shown in Figure 1 are text files, but in practice data can be held in a variety of formats. Data can be keyed directly into a wide range of software, it can be collected through devices such as optical marks readers which then transfer the information to a dataset, and it can result from the transfer of data held in one software package to another software package. SPSS has, from the outset, been able to import data hold in a variety for formats (see Norman N. Nye, C. Hadlai Hull, Jean G. Jenkins, Karin Steinbrenner and Dale H. Bent *SPSS Statistical Package for the Social Sciences.* 2nd Edition, McGraw-Hill, 1975, pages 41 to 56) but the simplest format, and it is the one assumed in the Guide, is where data are held in a single block, where each row is a case and each column a variable.

An SPSS file (.sav) can be read directly by selecting 'File' on the main SPSS window, and then selecting 'Open' and then 'Data'. Data can also be brought into SPSS directly from dBase, EXCEL and ACCESS data block files by selecting 'File' and then 'Open Database'. Text files with the suffix .txt, and comma separated value files with the suffix .csv, can be read in SPSS version 14 by selecting 'File' from the main SPSS menu and then selecting 'Read Text Data' from the resulting dropdown list. Data in free form format can be brought into SPSS using SPSS syntax and, while SPSS syntax is introduced in Section 24, importing data in free form format is not covered in this Guide.

If data are to be brought in from EXCEL, ensure that it follows the data block format exactly. Do not attempt to include subtotals, column subheadings, row titles and subtitles or other explanatory text. Data can also be exported from SPSS into a number of other formats.

The text (.txt) files in Figure 1 can be taken into SPSS comparatively easily (almost too easily – pitfalls await the unwary). DCSF have produced lists of NPD variables, with their associated codes, and these provide *a* point of reference when making decisions about the character of the data in the text files. There may be similar guidance for other datasets.

If this has not already been done, copy the text files to an appropriate folder on a local computer. The next steps require that researchers have already familiarised themselves with the content and meaning of the variables involved. However, while you know what to expect, SPSS does not. Work on the assumption that the first case in a variable, that is the cell immediately below a variable's name, provides SPSS with its cue as to the character and width of a variable. If the first case (cell) in a variable is blank, SPSS may well set the column width as 0 (zero). If you leave that width in place, any later instances of data, whether numbers or characters, in this variable will be lost during the transfer to SPSS. Additionally, if a string variable is read as a numeric variable there is, again, a serious risk of data being lost. In the world of education, for example, if the first case of a key stage 1 assessment record is 3 (i.e. the pupil achieved level 3 in that key stage 1 assessment) SPSS will classify the variable as numeric. This would be unfortunate, since the same variable will include outcomes at level W, 2A, 2B and 2C. These and any other alphanumeric records will be lost if they are imported into what SPSS has determined is numeric variable.

Additionally, if the variable width is less than the number of characters in the data to be imported, then that data will be truncated. The user who accidentally imports the text 'General Election' into a string *Election type* variable that is 10 characters wide will lose 'ection' from the record.

It is partly to account for these possibilities that so many variables shown in Figure 2 were taken into SPSS as (fairly long) string variables. While alphanumeric and ordinary text data are deleted if they are imported into numeric variables, all the characters in a numeric variable are retained if they are imported into an SPSS string variable of sufficient width. Converting these string variables back to numeric variables is a matter of moments, and is covered in Section 7 of the Guide.

Once the necessary homework on the variables in a dataset is complete, and, with SPSS open, select 'File' and 'Read Text data' from the menu at the top of the SPSS screen. Use the Browse facility to locate the text file needed and select it.

The user will be presented with the first of six 'Import Wizard' dialogue boxes.

Import Wizard 1. This will ask whether data match a predefined format. The default is 'no'. Since the SC can vary from one year to the next, leave that in place and click the 'Next' button.

Import Wizard 2. This will ask how variables in the file are arranged. The default position is 'Delimited', and that should be left in place. The Wizard will also ask whether variable names are included at the top of the file – select 'Yes' and click the 'Next' button.

Import Wizard 3. You will be asked which line number the first case begins on. The SC already

has variable names, and the window should show line 2. This can be changed as necessary. You will also be asked how cases are represented, and 'Each line represents a case' should be shown. Finally, you will be asked how many cases you want to import. The window should indicate 'All cases'. All being well, click the 'Next' button.

Import Wizard 4. This Window asks how data are delimited, that is what is it that separates one variable from another – select 'Tab'. Different files may well use other delimiters. You will also be asked what the text qualifier is. Select double quote and click the 'Next' button. Text delimiters can vary in the same way that Tab delimiters can vary.

Import Wizard 5. This window gives a view of the first few cases in a datasets, and you can check (review) at this point whether data have been allocated to the correct variable. If, for example, the variable 'Gender' is followed by the variable 'Age', and the age variable is populated with the values 'M' and 'F', there is a good chance that the wrong delimiters have been chosen during Import Wizard 4 (or that the data have been corrupted in the source file). Select the 'Back button, and try changing the delimiters.

However, all being well, Import Wizard 5 allows the user to determine the character and width of each variable and, in the case of numeric variables, to set the number of decimal places. This is where the user needs to take care to avoid losing alphanumeric data or truncating other data. For numeric variables, the numbers of decimal places also needs to be taken into account.

Remember the precautionary principle set out above. Unless absolutely certain about the character of all the records in a variable, import data as string variables with a column width of *at least* 10, depending on the variable in question. If there is uncertainty about how many characters there may be in, for example, the record of the language spoken by a child or of an adult's occupation, increase the variable's width. SPSS will allow string variables of up to 225 characters.

Select (click on) each variable in turn, and do not accept the default position without very good reason. When all the variables have been checked (reviewed), and only when all variables have been checked, click on the 'Next' button. If you click on the 'Next' button before you have completed the check on each variable, may find that you need to begin again.

Import Wizard 6. This gives the prompt 'Would you like to save this file format for future use'. The default position is 'No', and that has been accepted in this worked example. You will also be asked whether you would like to paste the syntax. (Syntax is discussed in Section 24). The default position is 'No' and that has also been accepted in this instance. You can now select 'Finish'. The text file will be imported into SPSS, and can be saved under an appropriate name.

Once the data have been brought into SPSS, save the file. You now have the option of running SPSS frequency tables for each variable to check whether data have been truncated, dropped or allocated to the wrong variable. (Bear in mind that it would not, for example make sense to run a Table of home postcodes from a national dataset with several million records.) Running frequency tables at this stage will provide a record of the source information, including a record of any codes used and of the number of cases with missing data. The SPSS output file can either be printed or save as a SPSS .spo file. Whether printed or saved as a .spo file, a record of the source data is well worth keeping at least until a project is complete. The procedures for running a frequency table are shown in the next Section.

5. Using Frequency Tables to check for missing data and miscodes, and to provide a record of source codes

Once a text file has been read into SPSS, look at the data given for the first case. You may be able to identify without further ado whether data have been truncated or scrambled and allocated to the wrong variable. If it is obvious that problems have occurred, make a note of where and what at the fault is. Then begin the file import procedure again, correcting for the error or errors that have been identified. Where errors do occur, they are likely to be because the wrong delimiters have been chosen or because data have been truncated.

Assuming that the visual scan does not identify any problems, you may wish to give variables labels appropriate to the project task in hand (and which will be meaningful to those reading output from the project. On the other hand, there is not a great deal of point spending time typing in variable labels if it turns out that the data have not been imported correctly, and that the whole import procedure needs to be re-run.

It is up to the user to decide whether typing in variable labels will help during the frequency table checking exercise. If the existing variable names are confusing as they stand, go to SPSS Variable View, and in the 'Label' cell on the same row as the variable name in question, type in a meaningful Variable Label. On the other hand, if a variable name is meaningful as it stands, leave it as it is at this stage, and at a later stage type into the 'Label' cell the name that is to appear in SPSS output.

Running a Frequency Table in SPSS is so simple that it is easy to overlook the crucial role it can play.

1. Select 'Analyze' from the SPSS main menu at the top of the SPSS window
2. followed by 'Descriptive Statistics' and 'Frequencies' from the dropdown menus which follow.
3. A 'Frequencies' window will follow, with a list of all variables in the dataset shown on the left. In the example in Figure 6, the variables shown have all been given labels.
4. Left click on the variable/s in interest and then
5. Select the arrow in the middle of the Frequency window. This should be pointing towards the 'Variable(s)' section of the dialogue box. Clicking on the arrow will transfer the name of the variables selected to the 'Variable(s)' section.
6. When all variables of interest have been transferred, click the 'OK' button.

For future reference, note the 'Statistics' Button in the 'Frequencies' dialogue box. This is referred to again in the Section on Visual Bander (Section 24). Buttons such as Statistics buttons appear in a number of dialogue boxes. They are not all discussed in the Guide, but are there for the user to investigate.

The Frequencies procedure is, as said, simple to use, and should not be overlooked. For a variable which has not been given value labels, it provides a list of the source codes. These have a use, and one of these is referred to in Section 10.

Frequency tables also show the total number of cases in a dataset. For example, frequency tables run on variables in the January 2006 and January 2007 pupil datasets show them to contain 7,669,115 and 7,622 pupil records cases respectively. Totals of that sort provided a check on the totals calculated in, for example, SPSS Crosstabs or SPSS Tables at a later stage. Totals that differ from those in frequency tables would need to be explained. If tables totals do not agree with the totals from Frequency Tables, would you really want to say 'I don't know' if that point is raised with you in the middle of a meeting? To avoid being caught out on either front, use the Frequency Tables facility. (There is a deliberate mistake included in a later Section Guide, where the total in a cross tabulation does not agree with the 2007 frequency table totals. The actions taken in that Section are set out on a step-by-step basis – they just do not include checking with a frequency table total.

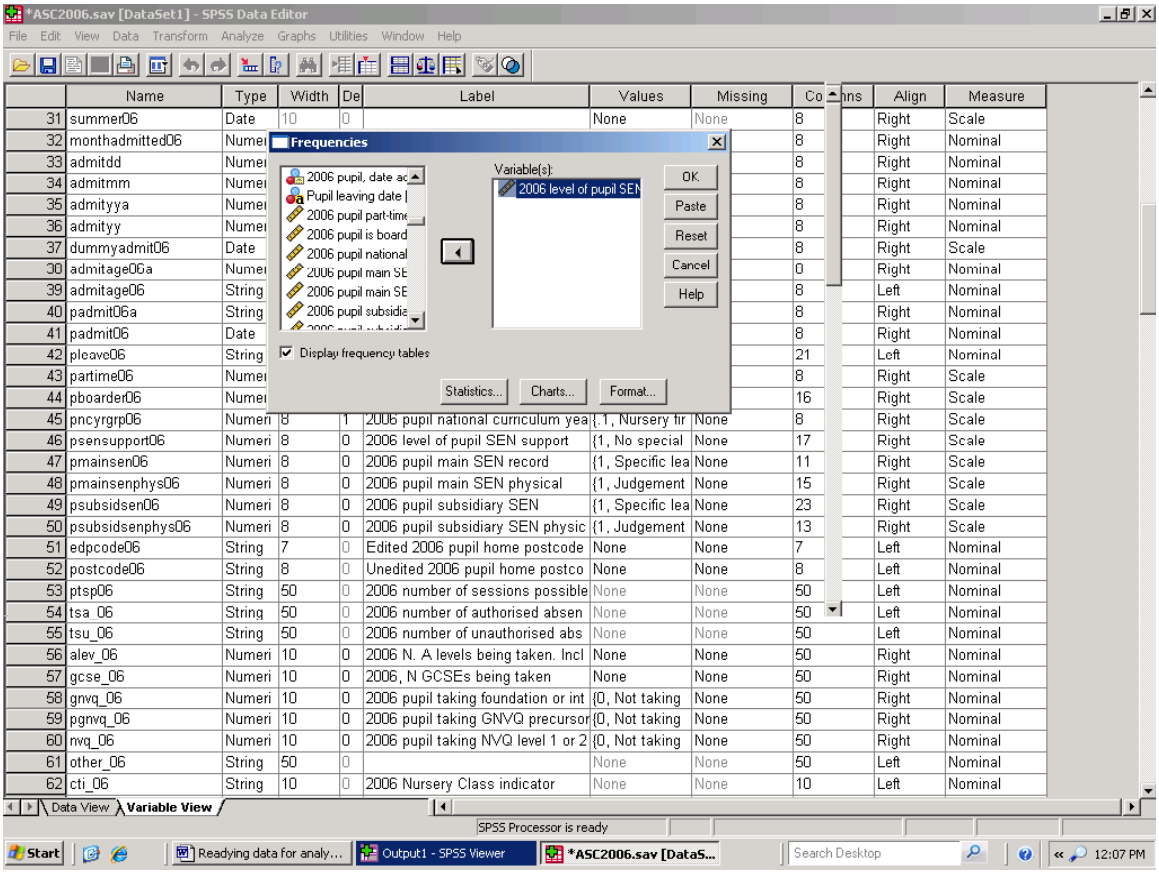
Total can differ because frequency tables include missing (blank) values, and some other procedures in SPSS do not. At the simplest level, researchers will no want to be caught out by failing to check for missing values by running a frequency table. Returning to the episode where someone questions a difference in totals, a general reference to missing values is unlikely to get our hapless researcher off the hook. For a sceptical audience, even the appearance of a lack of quality checks can bring the credibility of a report into question. If it is indeed the case that those quality checks are not in place, the scepticism is justified. Missing (blank) records can seriously distort statistical analysis.

One response to missing data is simply to delete the pupil records which have missing values in a key variable. In other instances missing data are equivalent to one of the actual codes used and can be recoded as such. Further, cases with missing data can be of interest in themselves, and this is touched on further in Sections 7 and 23.

Section 7 introduces facilities in SPSS for inferring missing data through statistical extrapolation. Ultimately the route taken reflects the research issues at stake, but missing data need to be dealt with.

On a final note, if there are a very large number of values in a variable this will lead to unwieldy frequency tables, which are time-consuming to produce and in some instances virtually impossible to use. Listing every case of a unique pupil identifier in a dataset with more than 7.5 million records would produce a monster of a frequency table. (Section 15 describes procedures for identifying the number of pupils with missing unique identifiers, and those procedures can be applied to other variables with too many values for the number of cases with missing (blank) values to be checked in a frequency table. Ground rules need to be followed sensibly, not slavishly.)

Figure 6. Running a Frequency Table – level of support for pupil special Educational needs



	Name	Type	Width	De	Label	Values	Missing	Co	Align	Measure
31	summer06	Date	10	0		None	None	8	Right	Scale
32	monthadmitted06	Numeri						8	Right	Nominal
33	admitdd	Numeri						8	Right	Nominal
34	admitmm	Numeri						8	Right	Nominal
35	admityya	Numeri						8	Right	Nominal
36	admityy	Numeri						8	Right	Nominal
37	dummyadmit06	Date						8	Right	Scale
38	admitage06a	Numeri						0	Right	Nominal
39	admitage06	String						8	Left	Nominal
40	padmit06a	String						8	Right	Nominal
41	padmit06	Date						8	Right	Nominal
42	pleave06	String						21	Left	Nominal
43	partime06	Numeri						8	Right	Scale
44	pboarder06	Numeri						16	Right	Scale
45	pncyrgrp06	Numeri	8	1	2006 pupil national curriculum yea	{1, Nursery fir	None	8	Right	Scale
46	psensupport06	Numeri	8	0	2006 level of pupil SEN support	{1, No special	None	17	Right	Scale
47	pmainsen06	Numeri	8	0	2006 pupil main SEN record	{1, Specific lea	None	11	Right	Scale
48	pmainsenphys06	Numeri	8	0	2006 pupil main SEN physical	{1, Judgement	None	15	Right	Scale
49	psubsidsen06	Numeri	8	0	2006 pupil subsidiary SEN	{1, Specific lea	None	23	Right	Scale
50	psubsidsenphys06	Numeri	8	0	2006 pupil subsidiary SEN physic	{1, Judgement	None	13	Right	Scale
51	edpcode06	String	7	0	Edited 2006 pupil home postcode	None	None	7	Left	Nominal
52	postcode06	String	8	0	Unedited 2006 pupil home postco	None	None	8	Left	Nominal
53	ptsp06	String	50	0	2006 number of sessions possible	None	None	50	Left	Nominal
54	tsa_06	String	50	0	2006 number of authorised absen	None	None	50	Left	Nominal
55	tsu_06	String	50	0	2006 number of unauthorised abs	None	None	50	Left	Nominal
56	alev_06	Numeri	10	0	2006 N. A levels being taken. Incl	None	None	50	Right	Nominal
57	gcse_06	Numeri	10	0	2006, N GCSEs being taken	None	None	50	Right	Nominal
58	gnvq_06	Numeri	10	0	2006 pupil taking foundation or int	{0, Not taking	None	50	Right	Nominal
59	pgnvq_06	Numeri	10	0	2006 pupil taking GNVQ precursor	{0, Not taking	None	50	Right	Nominal
60	nvq_06	Numeri	10	0	2006 pupil taking NVQ level 1 or 2	{0, Not taking	None	50	Right	Nominal
61	other_06	String	50	0		None	None	50	Left	Nominal
62	cti_06	String	10	0	2006 Nursery Class indicator	None	None	10	Left	Nominal

6. Inserting a new variable, setting its character and using the 'Compute' facility, selecting and deleting records, defining a set of variables

The merged 2002 2003 2004 2005 London Pupil Dataset (LPD) was established with computing capacity constraints in mind. However, it still contains variables from different datasets and, because the data are longitudinal, it also contains pupil records from different years. With over a thousand variables, appropriate Variable names and labels are needed to avoid total confusion. Variable names should be meaningful (and still be meaningful in six months time). Variable labels will appear in output, and should be meaningful to those who read it.

In the LPD some variables will refer to individual pupils, while others will refer to the school attended. One simple step is to prefix pupil variables with the letter 'p' and to end them with a year indicator, as with the edited pupil home postcode variable ppcode07. We will see in Section 12 that SPSS will not add a variable from one file to another, if a variable with the same name appears in both dataset. If we assume, for the sake of discussion, that pupil postcode appears as 'postcode' in files for 2007 and 2008, labelling one ppostcode07' and the other 'ppcode08' neatly gets round this problem, as well as providing accurate names for the variables. The variable list given in the Appendix may not appear at first glance to accord with this principle, since a large number of early variable have no year suffix. These variables are for 2002, the first year of the longitudinal dataset. It is precisely the absence of the year suffix that identifies them as records for 2002. Once again, ground rules need to be followed sensibly rather than slavishly.

School records from the separate EduBase national education institution file can be brought in to provide information on the school attended, including its postcode. Variables providing information on the school attended can usefully be prefixed with the letter 's' and also end with a year indicator, as in spcode07. That prefix shows that a variable refers to the school attended and the 'YY' information at the end of both variables indicates the year in question, as with the pupil records. It is best to name variables in this way as work proceeds, rather than trying to do that it as a single exercise when all data have been brought together (by which time confusion will have set in, and work will have to start again at the beginning).

If data from different datasets and different years are to be merged, it can be useful at this early stage to create a 'flag' variable as the first variable in the dataset and give it the value 1. Inserting an appropriately labelled new flagging variable as the first variable in each of the datasets being used, including those providing data for the main dataset, also provides boundary posts, indicating where a particular range of variables came from.

This example establishes a flag for variables from the 2006 SC files, and this allows the user to select records for analysis from that year simply and accurately. Inserting a new variable involves using the 'Edit' facility, and your route to it will depend on the version of SPSS being used. The procedures illustrated in Figures 7 to 10 and set out below refer to SPSS version 14, and assume that this is the first new variable to be created in the current working session. Procedures to insert a new variable may differ in other releases of SPSS.

1. left click on the name of the first variable in SPSS Variable View,
2. then click on 'Edit' in the SPSS main menu
3. and then select 'Insert Variable' form the dropdown list which follows.
4. A new first variable will be created 'above' that first variable and, in this instance, be given the name VAR00001.
5. Clicking on that name will allow you to change it to something more appropriate. Here it is 'flag06'. The 'term 'flag' is reserved for SC data, to avoid confusion with a flag for variables from another 2006 dataset.
6. Figure 7 shows the variable 'Type' cell immediately to the right of the new variable name, and on the same row. By default, the new variable is numeric. If you select that cell, you will be given the option of changing the variable to a string variable. However, on this occasion, leave that as it stands.
7. It is at this point that, in Section 7 below, you will change a string variable into a numeric variable. A later Section shows how to create a variable with a date format.
8. The cell to the right of that in the 'Width' column, again on the same row, allows the user to set the width of the variable. Left click on that, and set the width to 1.
9. The cell to the right in the 'Decimals column' allows the user to alter and set the number of decimal places for a numeric variable. Click on that cell and set the number of decimal places in this instance to 0 (zero).
10. On the same row, in the 'Label' cell, type in 'Flag for 2006 pupil SC record', which should prove useful for future reference.

11. To give each case in the flag variable the value 1, select 'Transform' in the SPSS main menu at the top of the screen, and select 'Compute' from the dropdown list which follows.
12. A 'Compute Variable' dialogue box, shown in Figure 9, will appear and in the white cell below the heading 'Target Variable', key in a short variable name, in this instance 'flag06' then.
13. Key the number 1 into the white cell headed 'Numeric Expression' and then
14. Click on the 'OK' button.

SPSS will now insert the number 1 in the flag06 variable for all records. Running a frequency table on flag06 will show how many 2006 records (or 'cases') there are in the dataset but, more importantly, anyone with longitudinal data wishing to analyse individuals with an SC record for a particular year, can run a straight forward 'Select if flag06=1' procedure, as shown in Figures 9 and 10.

But look carefully at Figure 10. This file is called NPD0607; it is the merged 2006 2007 dataset. The radio button 'Delete unselected cases' has been selected. Clicking on the 'OK' button at this stage would delete those 2007 records with no 2006 counterpart, which is not something you want to do by accident. Deselect that radio button and select 'Filter out unselected cases'. This will focus the computer's attention simply and effectively on 2006 records without your or someone else's work on 2007 records being destroyed.

The same principle applies to the assessment datasets. However, a variable name can only be used once. Where files are being merged, variables being attached to a main dataset will be disallowed if their name already exists in the main dataset. As a case in point, the flag for key stage 1 assessment records, which will be attached to SC 2007, is 'k1flag07'. Creating and naming variables is not difficult, but it can expedite a research project (or it can clutter a dataset).

When you use a 'Select if' command, and then choose either to filter out or delete records, that choice will remain in place until its is actively changed. For example, you have a gender variable where boys are coded as 'B' and girls are coded 'F'. and you wish to code these as 1 and 2 respectively in a new variable. SPSS can do that when asked, in appropriate SPSS-speak, to give the new variable the value 1 when you have selected records so that only those with the gender code 'B' are considered, and that to give the new variable the code 2 when you have selected records so that only those with the gender code 'G' are considered.

Figure 7. Computing a single value for all cases – flag06

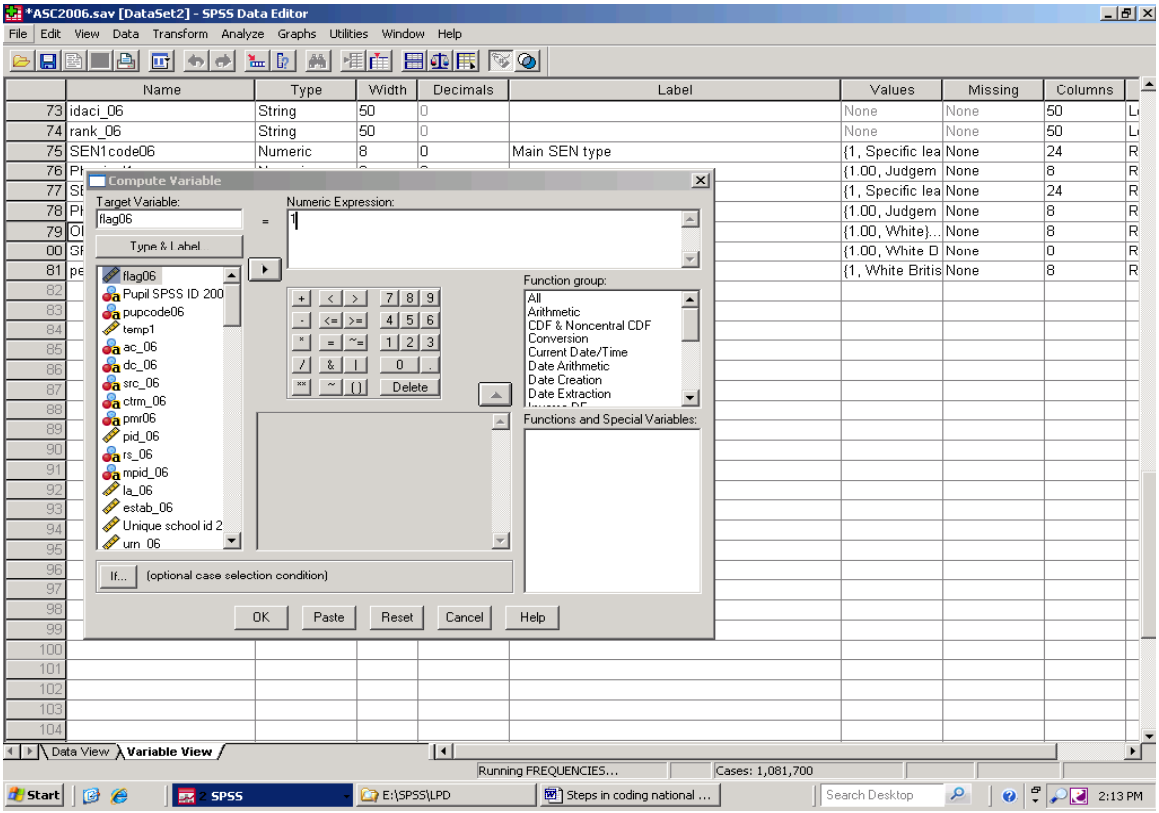


Figure 8. Selecting a subset of variables to analyse – step 1

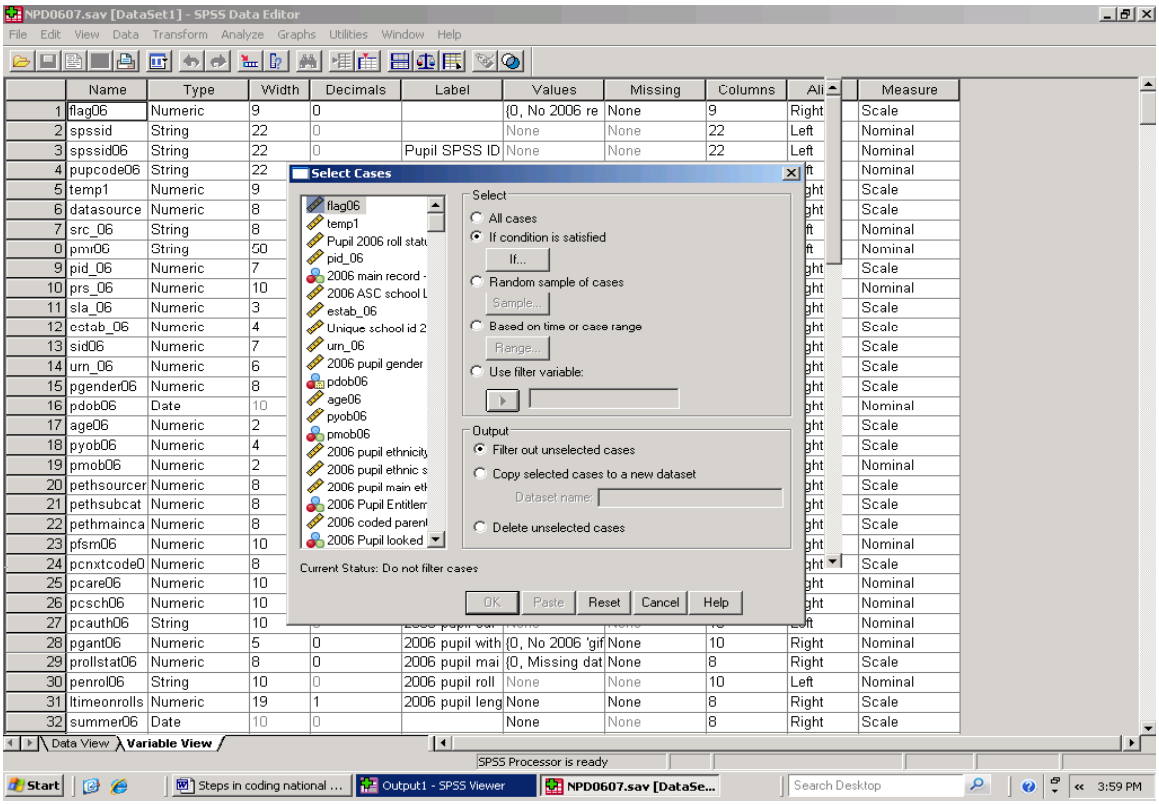


Figure 9. Selecting a subset of variables to analyse – step 2

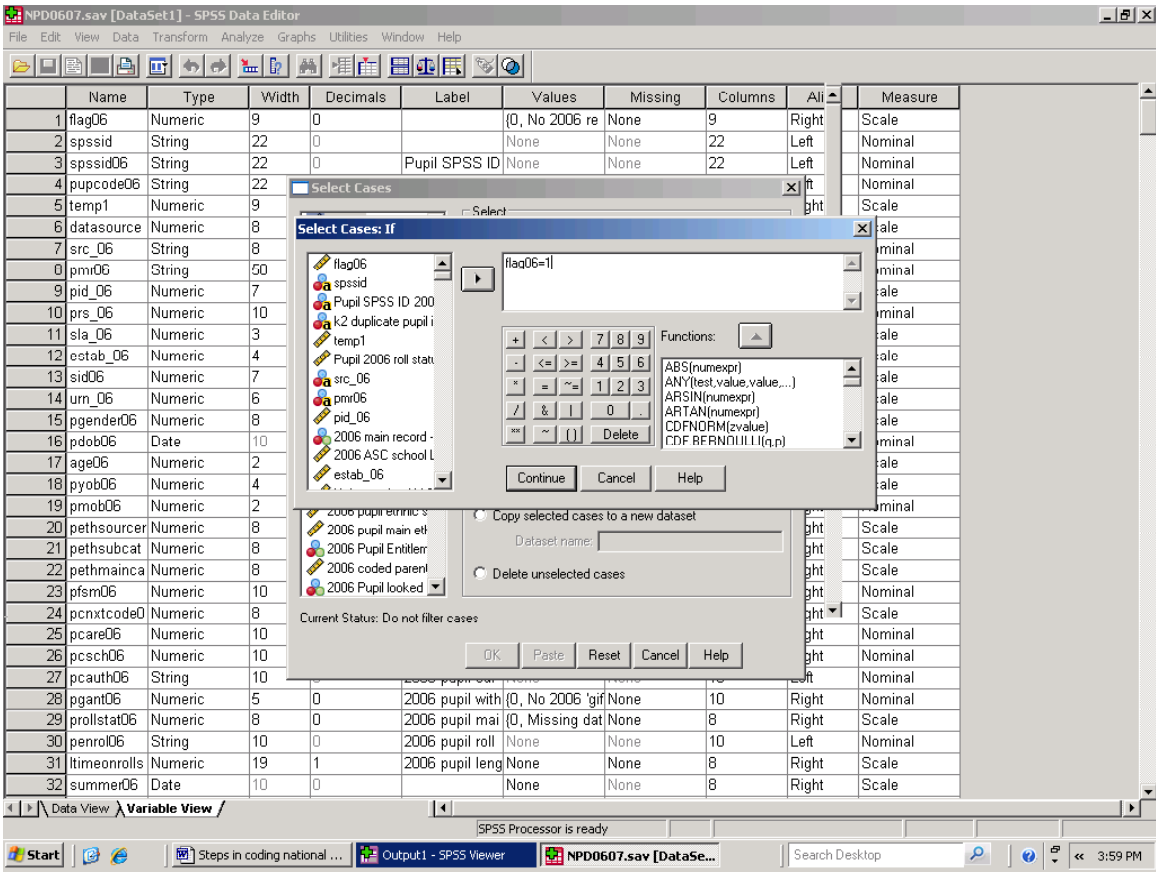
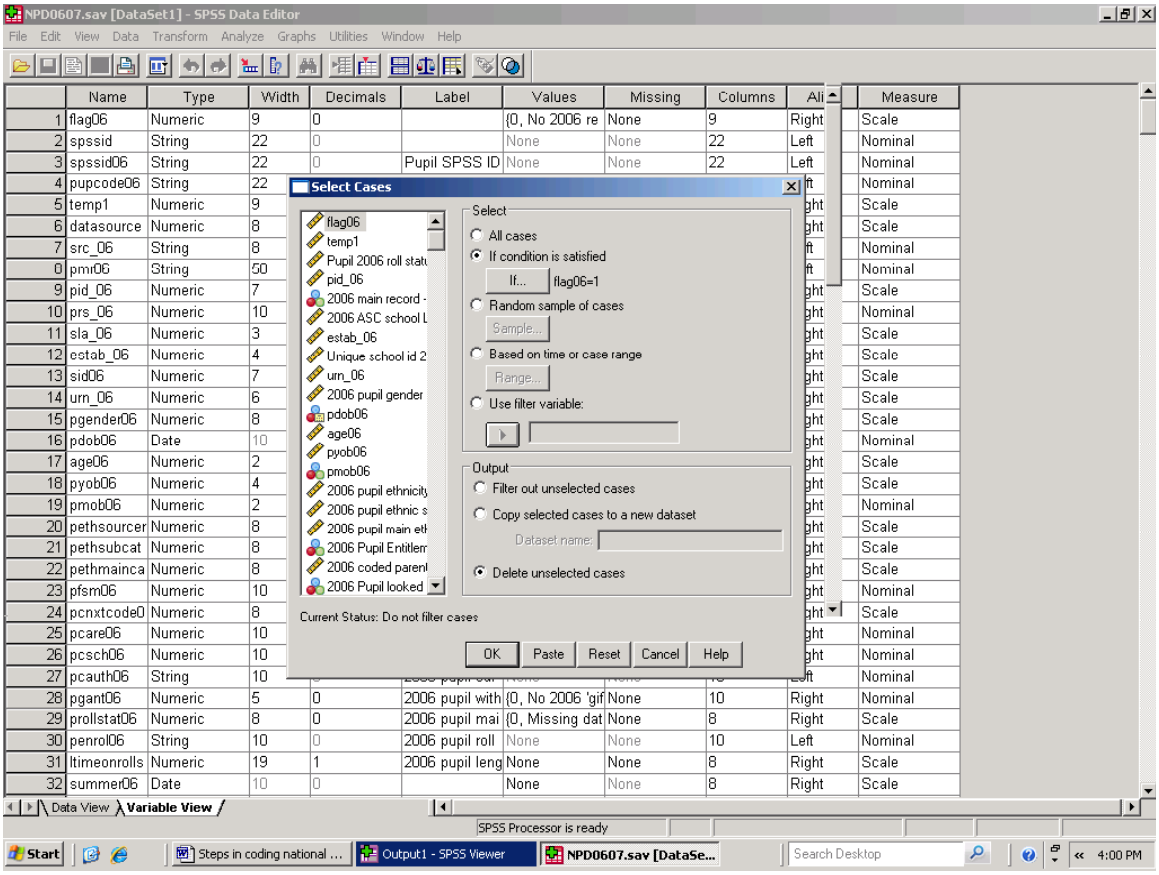


Figure 10. Selecting and deleting records – READ THE TEXT RELATING TO THIS FIGURE CAREFULLY



The exact steps involved in the 'Compute if' type of exercise are set out later, but the point here is that the second step was to filter out all records for boys. If carried out at this point, an analysis of attainment in a boys' secondary school would be decidedly odd. Records that have been 'filtered out' will continue to be filtered out until the 'All cases' radio button in the 'Select Cases' dialogue box is selected.

Where you have used the steps shown in Figures 7 to 10 to select a subset of cases to work on, ensure that the 'If Condition', 'Filter out' and/or 'Delete' radio buttons in the 'Select Cases' window are deselected as appropriate as soon as the work with that subset of cases is done. There are other ways of selecting a subset of cases. Regardless of which 'select if' approach is taken, switch it off as soon as work with that subset of data is completed.

All being well, having read Sections 5 and 6, you will now be at home with the steps involved in

- bringing text files into SPSS in a form which ensures that data are not lost
- running frequency tables which show datasets totals and the incidence of missing (blank) values in variables
- establishing and keeping a record of any codes used in a variable that has no value labels
- creating new variables
- establishing meaningful variable names and variable labels
- determining whether a variable will be a numeric or a string variable
- setting the number of decimal places in a numeric value
- computing values for a new variable
- Selecting and deleting records

7. Converting a string variable to a numeric variable, coding data as they stand, using the SPSS 'Missing' column to identify several missing value codes, and the Missing Values module

Missing data can be very revealing. DMAG Briefing 2008 - 27, for example, provides evidence of children missing the last year of compulsory education, and links those cases to missing key stage 3 records and to social disadvantage. Children with missing data (or completely missing roll records) are of interest in their own light, and a copy of DMAG Briefing 2008 – 27 is available on request.

If you are working in a local authority, one of your objectives will be to minimise the incidence of missing data. Data entry restrictions may be in place which disallow blank or inappropriately coded records to be created. In other instances, you may also have arrangements for referring incomplete records back to those who provided the data. Where neither of these apply, it may still be possible to replace missing data directly or after by triangulating one variable with another.

While there are caveats as to how far it 'captures' all pupils who live in poverty, entitlement to free school meals (FSM) is a frequently used measure of poverty. There are two codes in the FSM variable in pupil files released by DCSF: 0 (zero) indicates that a pupil is not eligible for free school meals, and 1 indicates that a pupil is eligible for free school meals. In work at the GLA, FSM code 1 has been given the value label 'Entitled to FSM'. All other pupils have been given FSM code 0, with the value 'No record of entitlement to FSM'. A first step in giving different values different labels is set out below.

1. Change the FSM variable from a string to a numeric format, with no decimal places, following steps 6 and 7 in Section 6.
2. Label the FSM variable as 'Pupil free school meal entitlement', and include in the title the year in question, for example '2006'.
3. Click on the values cell for this variable, which will take you to the window shown in Figure 11.
4. In this instance the value 0 has already been typed into the 'Value' section of the dialogue box, and the text 'No record of FSM' has been keyed into the 'Value' section of the dialogue box, and 'No record of FSM entitlement' has been keyed into the 'Label' section immediately below that.
5. The 'Add' button below the word 'Label' has then been 'left clicked', and that code and label have been added to the lower section of the dialogue window.
6. FSM code 1 can now be given the label 'Entitled to FSM' following the same steps.
7. Selecting the 'OK' button in the 'Value Labels' dialogue box applies the labels to the dataset.
8. Running a frequency table on the FSM variable will show whether there are any missing values.
9. *In this case* a missing value is equivalent to 'No record of Entitlement to FSM, and any missing data can be recoded as 0.
10. Select 'Transform' from the SPSS main menu at the top of the screen, and then select 'Recode', followed by 'Recode into the Same Variables' from the dropdown lists which follow.
11. Having taken those three steps, you will be shown the dialogue box in Figure 12.
12. The dataset variables are listed on the left of the dialogue box. Scroll up or down this list as necessary to locate the FSM variable. Left clicking on the FSM variable will highlight it.
13. The 'arrow' button to the right of the variable list should be pointing to the right. As long as it is, select that button, and the name of the FSM variable will be transferred to the 'Variables' pane to the right.
14. Left click on the 'Old and New Values' button below the 'Variables' section, and the 'Recode into Same Variables: Old and New Values' window shown in Figure 13 will follow. That Figure shows that the 'System or user missing' button on the left of that window has been selected and
15. 0 (zero) has been keyed into the 'New Value' 'Value' pane.
16. Clicking on the 'Add button' next to the 'Old -- > New' pane has added the code for missing data.
17. The next step is to click on the 'Continue' button at the bottom of the 'Old -- > New' pane, which will return you to the 'Recode into the Same Variables' window.
18. Once there, click on the 'OK' button

Figure 11. Value labels

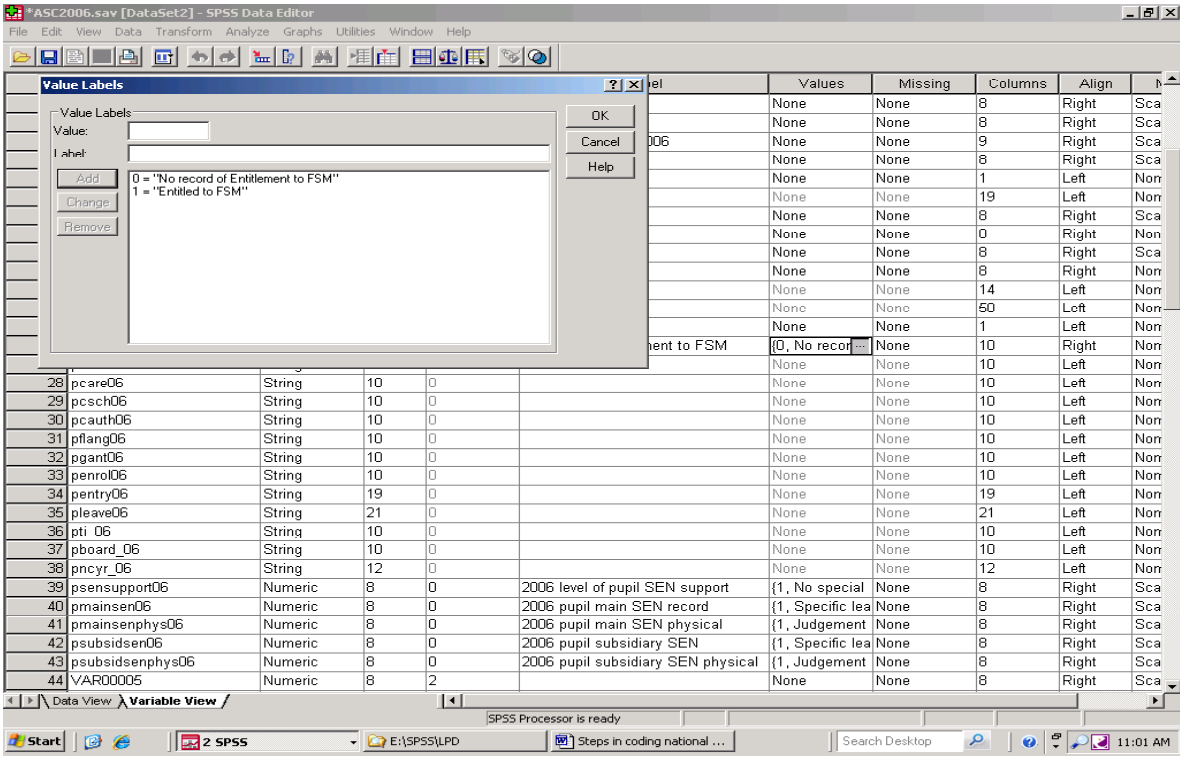


Figure 12. Coding missing values in a numeric variable –1

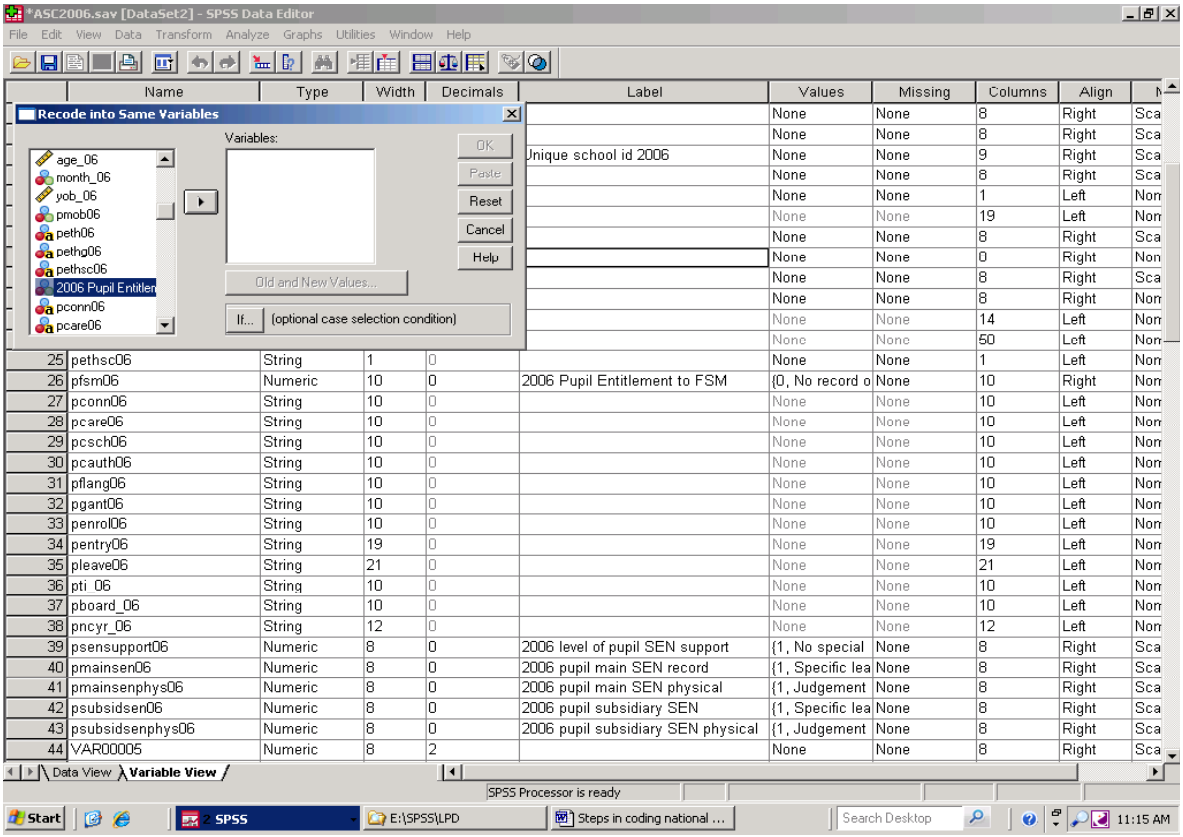
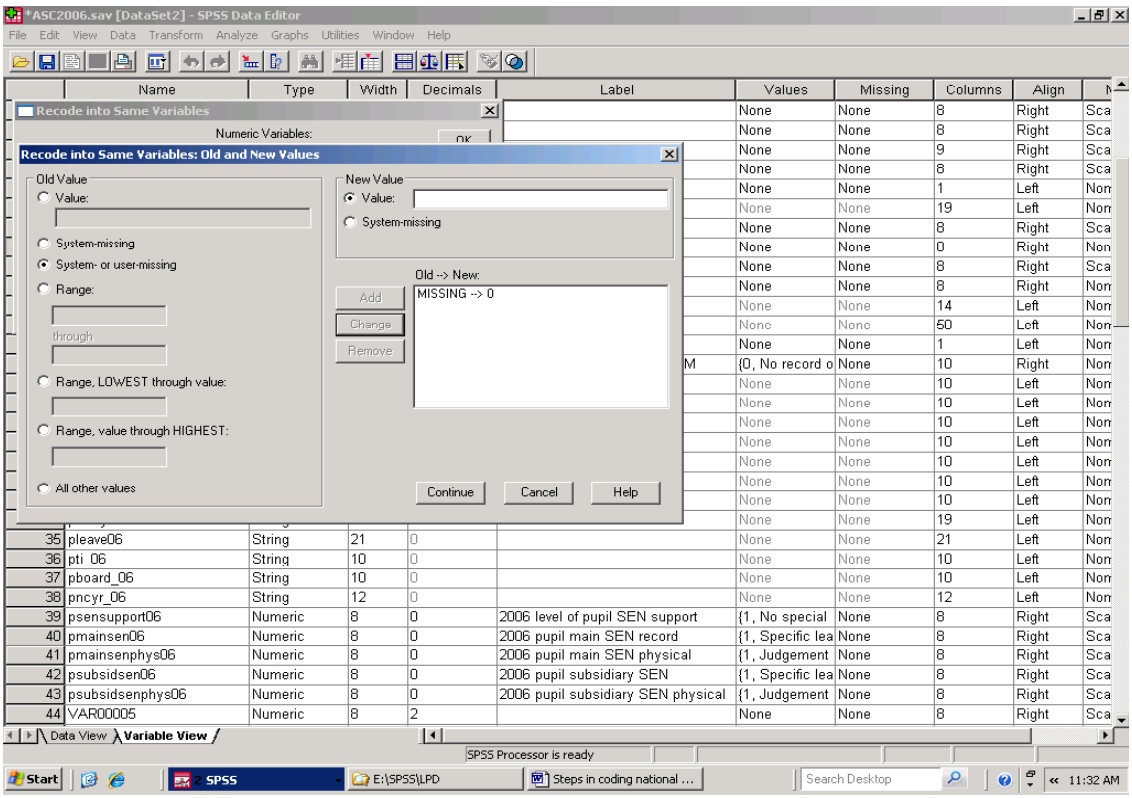


Figure 13. Coding missing data in a numeric variable – 2



The FSM variable was coded in this way with one eye on establishing when, and how often, pupils were entitled to FSM over the period 2002 to 2005. Figure 50 in Section 15 provides a longitudinal view of free school meal entitlement. It rests on a derived variable, created through a series of conditional 'Compute if' procedures which created different values for the groups of years when a pupil was entitled to FSM. Additionally, because FSM entitlement has been set to 1 for those who have a record of entitlement, the latter is simply the addition of the FSM variables for each year. It is a straightforward matter, but neatly illustrates the point that analytical needs, rather than happenchance, properly determine what codes are used.

It has been convenient here to have two codes for free school meal eligibility, one of which includes both pupils with the code 0 and those with no FSM record of any sort. In the 2008 'PLASC' extract, five pupils had no record of any sort in the age variable. In a numeric version of the variable, they could have been given a code of their own, for example 99, with the label 'Missing data', using the steps just described. While this is appropriate for simple tabulations of nominal variables, as a rule of thumb do not use this approach if a variable is to be included in statistical procedures the assume a level of measurement other than nominal. You will know that 99 is a flag for missing data in the age variable, but SPSS will not, and calculating a regression coefficient for pupil height on pupil age would be compromised if a large number of cases had been given the code 99, and were included in the calculation.

What SPSS recognises as missing values are, as we have seen, usually omitted from SPSS calculations, and there is an option which builds on this by coding missing values in a way that SPSS recognises as a missing values, and which excludes those cases from statistical calculations.
.

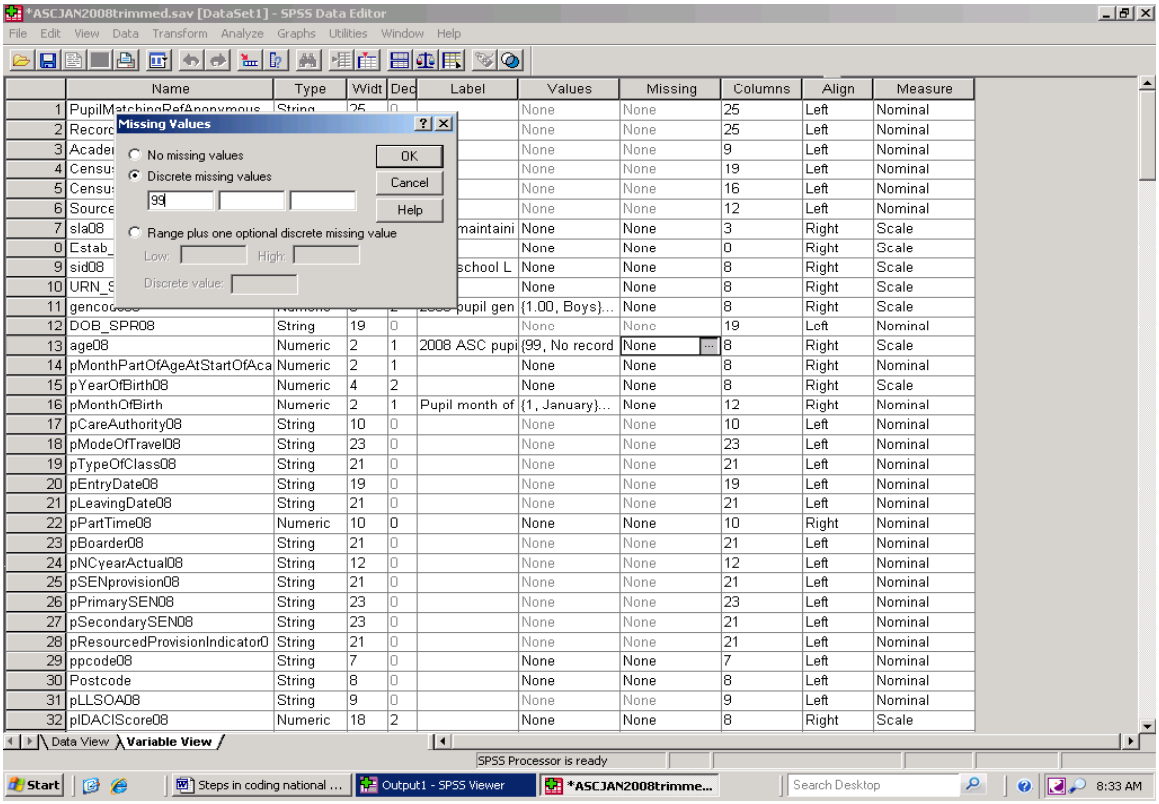
Figure 14 shows a SPSS 'Missing Values' dialogue box close to the variable 'age08'. That variable is the 13th in the dataset, and the 'Missing' cell for the age variable is highlighted. Selecting a variable's 'Missing' cell calls up the 'Missing Values' dialogue window shown in Figure 14, and this allows for up to three missing values for the variable in question. In this instance, the single value 99 has been keyed in. This alternative route gives the same missing value number as before, but SPSS will now recognise it as a missing value and exclude cases with that value from analysis.

The option of including three missing value codes reflects the reality that data may be missing for more than one reason. In survey returns data may be 'missing' because the respondent had no view, because he or she objected in principle to answering a question, or because the question was not asked. Since you cannot know

which of these applies if the record is simply blank, you are only likely to use this multiple missing number option if a dataset *already* contains multiple values for missing data, that is if missing data are not actually missing in the sense of being blank.

.

Figure 14. Coding Missing Values in 'Missing Values' dialogue box



A further way of handling missing values involves using statistical techniques to estimate what missing values should be (what you will be able to do will depend on the software installed in the machine you are using). Tables 15 and 16 show two early steps in replacing missing values by selecting 'Transform' on the main SPSS menu, and then 'Replace Missing Values…' from the dropdown list. This approach provides the option of replacing missing values with any of, the series mean, the mean of nearby points, the median of nearby points, or through linear interpolation or by linear trend. The explanation provided in the SPSS help file is shown below between the quotation marks.

*"**Series mean**. Replaces missing values with the mean for the entire series.*
***Mean of nearby points**. Replaces missing values with the mean of valid surrounding values. The span of nearby points is the number of valid values above and below the missing value used to compute the mean.*
***Median of nearby points**. Replaces missing values with the median of valid surrounding values. The span of nearby points is the number of valid values above and below the missing value used to compute the median.*
***Linear interpolation**. Replaces missing values using a linear interpolation. The last valid value before the missing value and the first valid value*

after the missing value are used for the interpolation. If the first or last case in the series has a missing value, the missing value is not replaced.
***Linear trend at point**. Replaces missing values with the linear trend for that point. The existing series is regressed on an index variable scaled 1 to n. Missing values are replaced with their predicted values."*

The first explanation is straight forward, but the next two options may well need further explanation. Figure 16 to 17 shows the dialogue box in which replacing missing values by the median of nearby points has been selected. Additionally, a variable has been selected from the list on the left and transferred to the 'New Variable(s)' section, and SPSS has automatically provided a new variable name to take the recoded information. The 'Span of nearby points' information is set by default as number and 2 points, and has been changed here to 999.

In this instance, the record of pupil age will become the median age value of the 999 cases

on either side of a case where the value is missing. This is not a sensible thing to do if those 999*2 cases are where they are for reasons which that simply nullify the exercise. Sorting pupils by age would be a case in point – there would be no cases 'above' the first instance of a pupil with a missing age record. Sorting the dataset on a numeric version of pupil national curriculum year (which would lose those pupils in reception and nursery classes, who have NC year group codes N and R) *beforehand* and then using the median age approach for pupils in national curriculum year group 2 and above would make some sense, since a pupil's national curriculum year group is largely (but not always and everywhere) age dependant. There should be a *good* reason for choosing a particular statistical method for imputing missing data.

Figure 15. 'Transform' and 'Replace Missing Values'

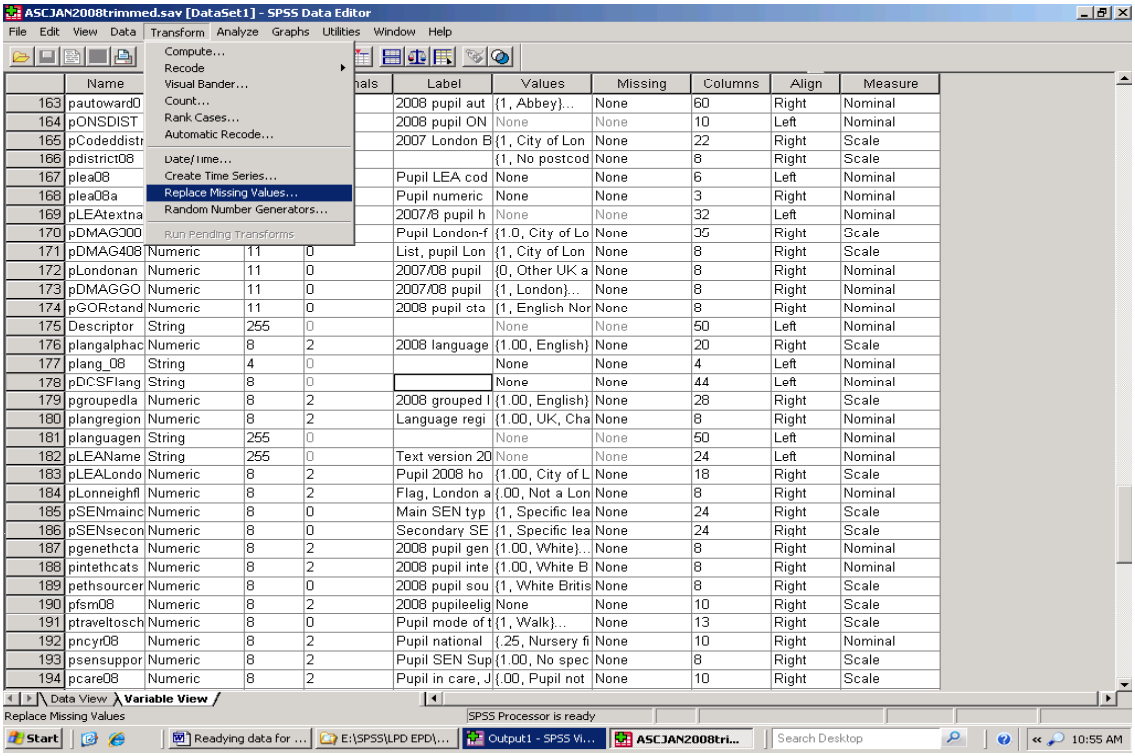


Figure 16. Options for replacing missing values

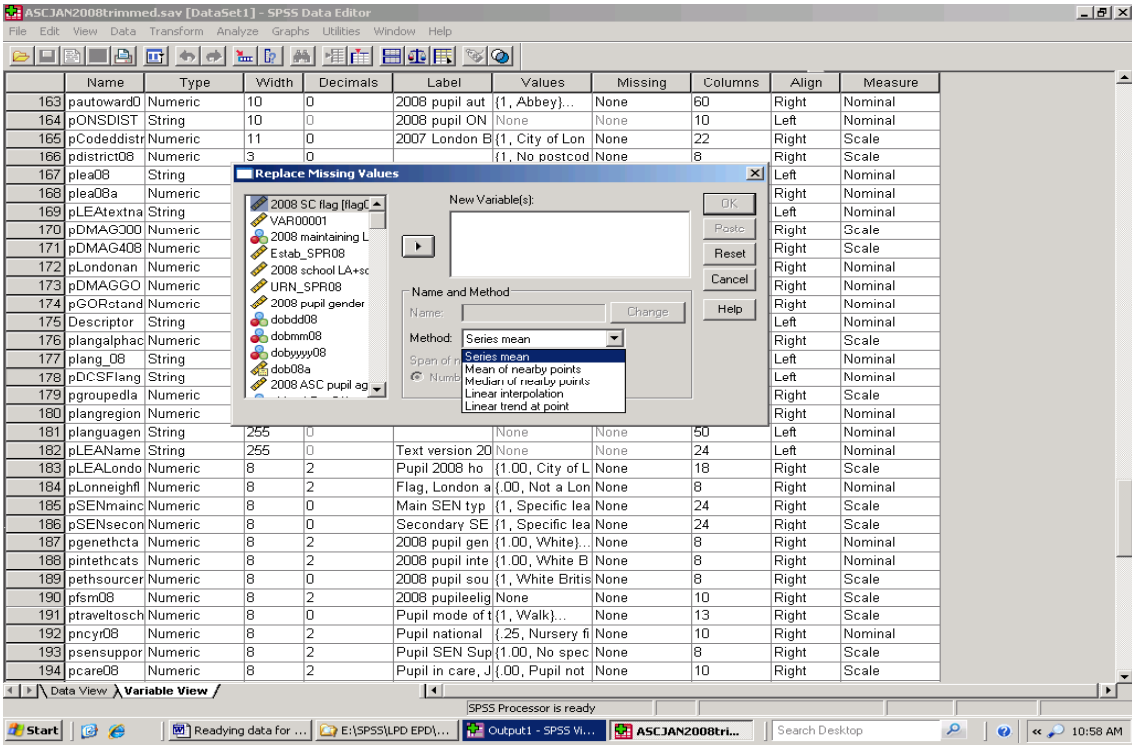
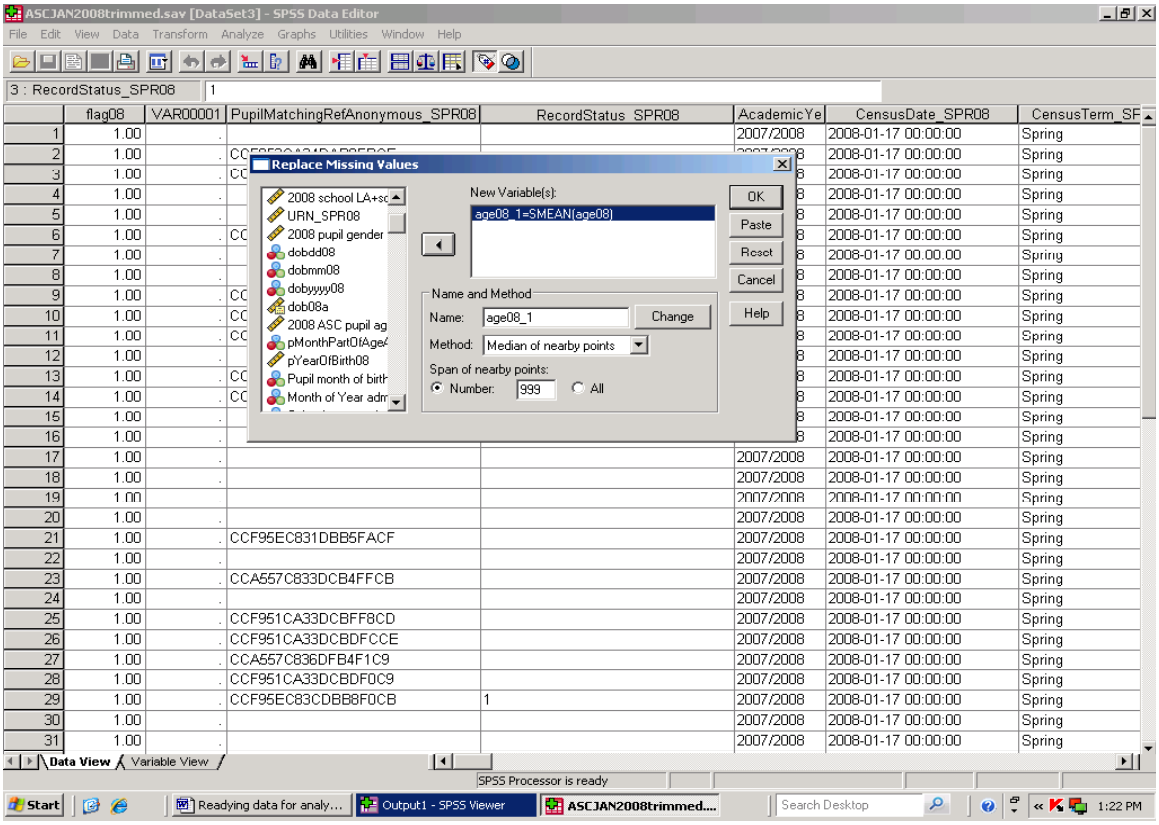


Figure 17. Replacing missing values with the median of nearby points



If there is no reason for imputing a particular missing value, then the research analyst is probably better off triangulating cases with missing data with information from elsewhere (such as the national curriculum year group variable), and then using the 'Compute' or 'Recode' facilities to plug gaps in the dataset. The last two options are linear interpolation and linear trend. These are potentially useful with time series data. There is also a separate SPSS 'Missing Values' module, which offers more advanced statistical approaches to dealing with missing values. The SPSS Help file describes the 'Missing Values' module as performing "three primary functions:

- Describes the pattern of missing data: where the missing values are located, how extensive they are, whether pairs of variables tend to have values missing in different cases, whether data values are extreme, and whether values are missing randomly.
- Estimates means, standard deviation, covariances, and correlations using a listwise, pairwise, regression, or EM (expectation-maximization) method. The pairwise method also displays counts of pairwise complete cases.
- Fills in (imputes) missing values with estimated values using regression or EM methods."

If it is available on the computer you use, the module can be accessed by selecting 'Analyse' on the SPSS main menu, and then selecting 'Missing Value Analysis' from the resulting dropdown list as in Figure 18. This will open the window shown in Figure 19, where you can begin to exercise the choices this module offers. Interestingly, Figure 19 makes it clear that the approach is not restricted to continuous variables, such as GCSE or Key Stage point scores, but also applied to categorical variables (aka nominal and ordinal data).

The analysis of trends in SPSS and SPSS' logistic regression, nominal regression and probit analyses procedures also all involve facilities for handling missing data. These are not covered in detail in this Guide, and anyone new to issues associated with missing values who is considering using the more advanced facilities in SPSS should ensure that relevant statistical texts have been read in advance.

For those who have not covered this field, a short introduction, with some good pointers, is given in pages 62 to 72 in Barbara G. Tabachnick and Linda S. Fidell 'Using Multivariate Statistics' (Pearson International Edition, Fifth Edition, 2007). For a more detailed account, see Paul D. Allison 'Missing Data' Sage University Papers Series on Quantitative Applications in the Social Sciences, series number 07-136, Thousand

Oaks: Sage (2001). The ESRC Research Methods programme has sponsored a relevant website, which can be reached at the time of writing at **http://www.restore.ac.uk/PEAS/imputation.php**

and this provides one starting point. The website focuses on imputation and missing values, and also contains a number of useful links.

Figure 18. Opening the Missing Values module

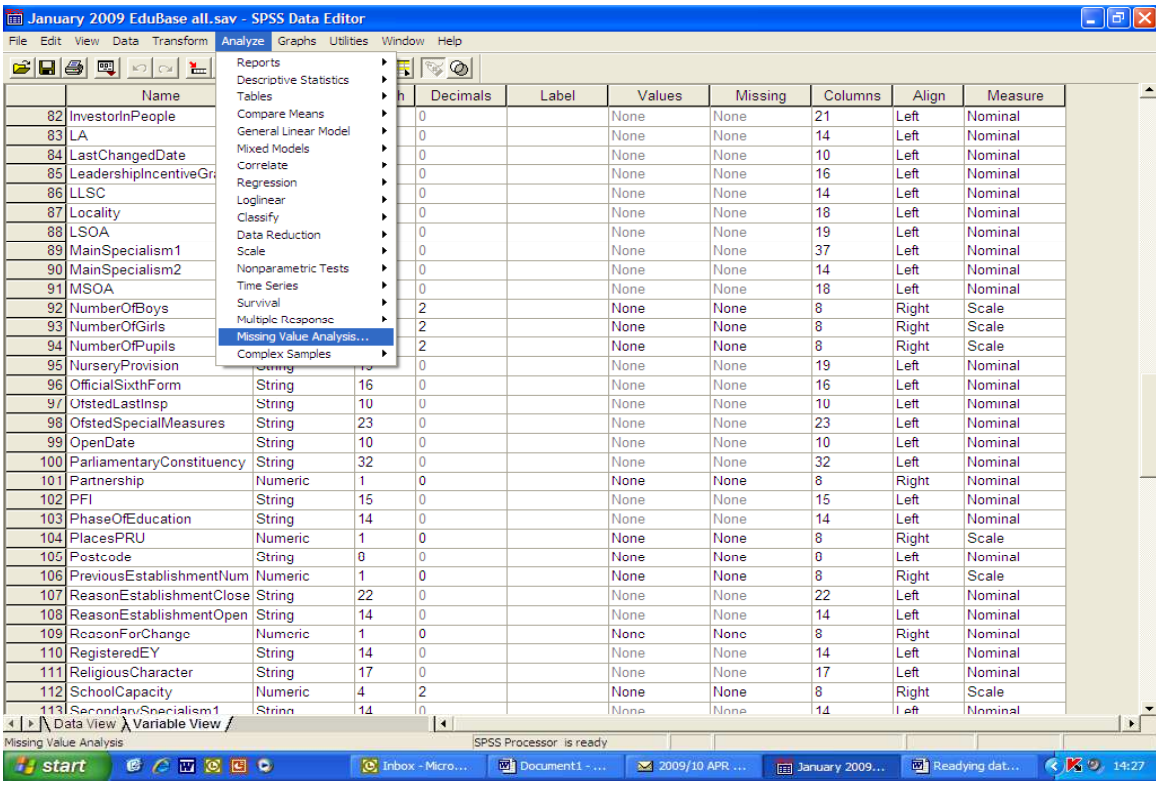
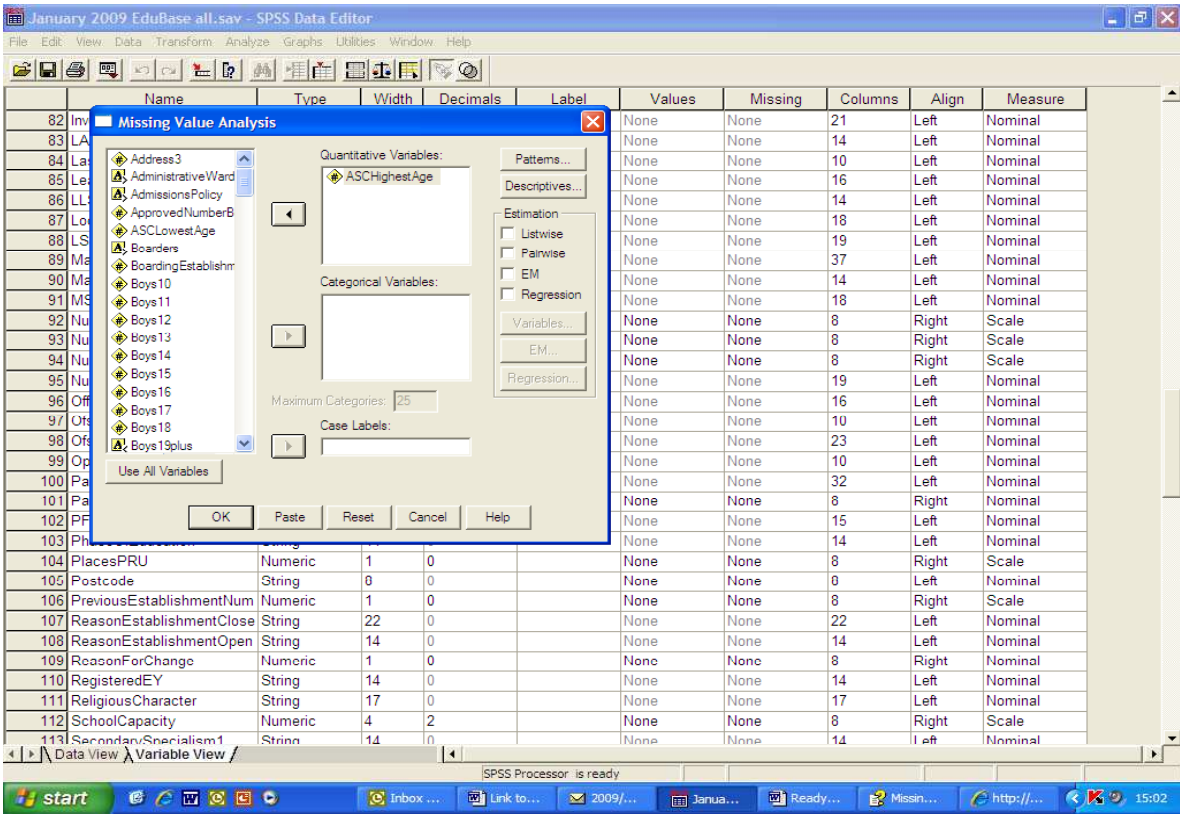


Figure 19. The 'Missing Value Analysis' window



8. Conditional 'If' statements and recoding data into a different variable – level of SEN support. Checking recoded data with Crosstabs

As a rule of thumb, assume that SPSS prefers to work with numbers. Where variables are taken into SPSS as string variables, there is an incentive fort change. FSM is an example of a variable that can be used as a numeric variable, almost as it stands, but a number of other string variables in the 'PLASC' files require a different approach. In these instances a wholly new numeric equivalent is created, and once these numeric equivalent variables are checked, source string variables are deleted from the working file to limit file size. A full copy of the source dataset is kept separately.

The worked example in this Section creates a numeric equivalent to the variable *sen_07* from the 2007 'PLASC' file.

Original Code	Meaning	Numeric SEN codes to be created
N	No record of special provision	1
A	School action	2
P	School action plus	3
S	Statement of SEN	5

This refers to the level of support for pupils with special educational needs (SEN). It is a string variable with comparatively few codes. These are shown on the left above. Their meaning is shown to the right, and the numeric code to be created is shown to the right of that. Missing (blank) data is to be coded as 1. The value label, which will also be created will be the meaning as shown. The 'Compute' facility referred to in Section 6 provides a straightforward, though potentially time-consuming, way of creating coded numeric equivalents to the source data.

1. Left click on the name of the variable immediately below the point where you wish to insert the new numeric variable. This may be immediately below the original string version, that is *sen_07*. If this is the first variable to be created in the current working session, SPSS will by default create a numeric variable eight characters wide with two decimal places called VAR00001.
2. Left click on that name, and type on the name of the new variable, shown here as 'sensupport07'.
3. Set the decimals to 0 (zero) by either typing 0 into the decimals cell in Variable View or by using the 'up' and 'down' arrows in the Decimals cell.
4. In the Label cell for the variable write '2007 level of SEN support'.
5. Left click on the Values cell for the variable – a Value label window will appear. In the Value Label dialogue window type the number 1, and for the label key in 'No record of special provision', then click on the 'Add' button. Repeat this procedure until all the new numeric codes and their labels have been added.

As noted, a variation of the 'Compute' procedure set out in Section 6, will allow the user to create a numeric equivalent, but are not advised to take that route. However, that variation would require the selection of 'Transform' from the SPSS main menu, followed by 'Compute' with an associated conditional 'If' statement. Figure 7 in Section 6 shows the 'Compute Variable' dialogue box. Type in the name of the variable you wish to create followed by the first value you want to be entered. In this instance you would type in 'sensupport07=1'.

The 'Compute Variable' dialogue window has an 'If' button near the lower left hand corner. Selecting the 'If' button opens a 'Compute Variable If Cases' dialogue box. At the top of this box, and slight to the left of centre, there are two radio buttons. The first is 'Include all cases' and the second is 'Include if case satisfied condition'.

Selecting the second radio button will allow you to key in sen_07="N" as a conditional 'If' statement in the area immediately below the button. The double quotes are needed to let SPSS know that it is looking at text data in the original SEN variable. Figure 75 in Section 19 shows an (only slightly) more complex 'Compute Variable: If Cases=" " conditional statement.

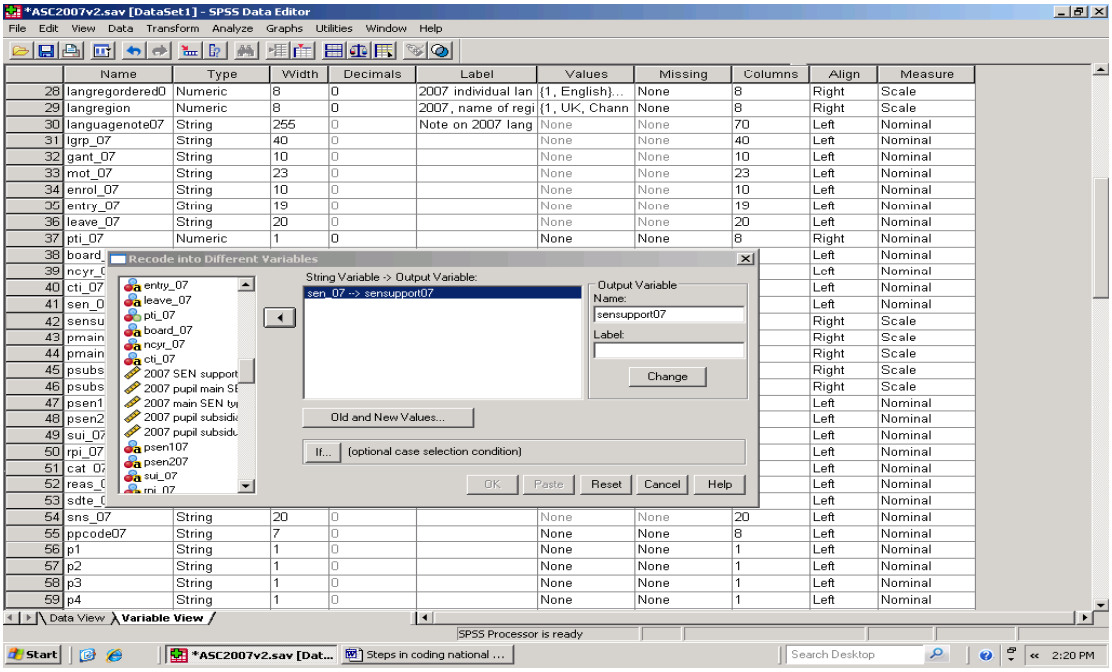
The user would need carry out separate 'Compute if' exercises for each of the numeric codes to create a fully equivalent numeric sensupport07 variable. There are times when separate 'Compute' exercises of this type have to be carried out and that can, as noted, be time-consuming, particularly if you are working with a large dataset.

However, a quicker option is available in this instance. The SPSS' 'Recode into a Different Variable' procedure will do most (but not all) of the recoding in a single step. To use this procedure select 'Transform' from the main SPSS menu and then select 'Recode' followed by 'Into Different Variables' from the next dropdown list. You will then be shown the 'Recode into Different Variables' dialogue box, which is illustrated in Figure 20.

A list of the variables in the dataset is shown in the left of the 'Recode into Different Variables' dialogue box. This is a common view of variables presented in SPSS dialogue boxes, and will be seen again. To the right of this an arrow button should at this stage point towards a section headed 'Input Variable -> Output Variable'. In this instance the variable sen_07 contains the source text codes: it is the 'Input Variable'. Select the name of that variable from the list on the left, and use the arrow between the two panes to transfer it to the pane on the right. (Note that in Figure 20 the arrow button points back towards the variable list. Clicking on that button would transfer the input variable back to the variable list.).

Figure 20 also shows that this dialogue box has an 'Output Variable' section with 'Name' and 'Label' sections immediately below. In the section headed 'Name' key in the name of the variable you wish to create. In this worked example the variable is 'sensupport07', which is a variable you may already have created using the 'Edit/Insert Variable' facility. Click on the 'Change' button below the 'Output Variable' section, and you will be prompted with the observation that a variable with that name already exists (which it should. It was deliberately created in at a particular point in the dataset in Step 2 above.) Keep variables in datasets in logical order. It makes finding a variable for that much easier. See Section 12. Accept the SPSS prompt, and click on the 'Old and New Values' button shown in Figure 20. This will take you to a new window, which is shown in Figure 21. (That Figure refers to the 2006 SC, which is shown here for purposes of illustration.)

Figure 20. Recoding into a different variable - 1



After this, type a source text code into the 'Old Values Value' section, and type its new numeric equivalent in the New Value pane. Click on the 'Add' button. The text code will be added to the 'Old-- > New' section, along with the new numeric equivalent. Repeat that procedure until all source codes with their numeric equivalents are listed. Note that the first code listed in Figure 13 is 'MISSING -> 1', which implies that pupils with no string record of SEN support are to be given the same numeric record as those with the string record 'N'. Once the old and new codes are entered, select the 'Continue' button at the bottom of the dialogue box. You will be returned to the 'Recode in Different Variables' window. Click on the 'OK' button in that window and the procedure will be run. Once the new variable has been created in the label cell for that variable key in '2007 SEN support'. This label will be shown in SPSS output, and should add meaning to that output. It will also be shown in the variable list in the exercise which follows.

One way of checking whether the text SEN codes have been properly recoded into the new numeric variable is to use the SPSS Crosstabs facility. To do this, select 'Analyze' from the main menu, and then select 'Descriptive Statistics' and 'Crosstabs' from the dropdown lists which follows. This will produce the dialogue box shown in Figure 22. The layout of the dialogue shows what will by now be the familiar list of dataset variables on the left, separated by an arrow button from other sections on the right. Select the source variable (sen_07) and transfer this to the 'Row(s) section in the usual way. You will see that the new variable is

listed under the Variable Label it has been given ('2007 SEN support'). Select that, transfer it to the 'Column(s)' section in the usual way and then left click on the 'OK' button.

Figure 21. Recoding into a different variable - 2

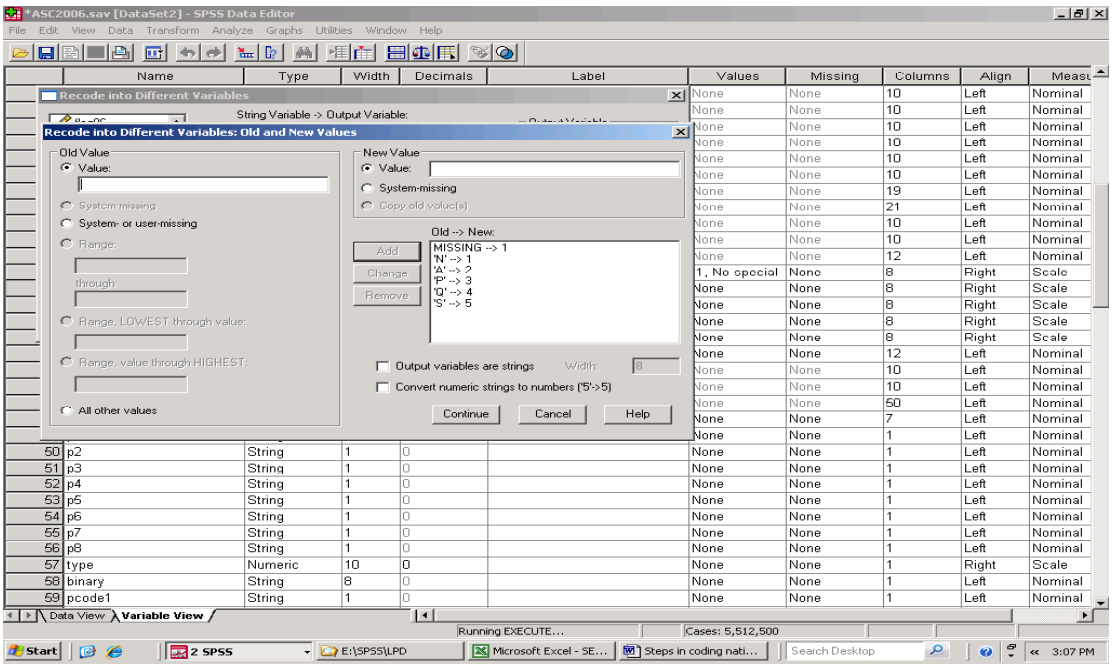


Figure 22. Tabulating string and numeric variables - Crosstabs

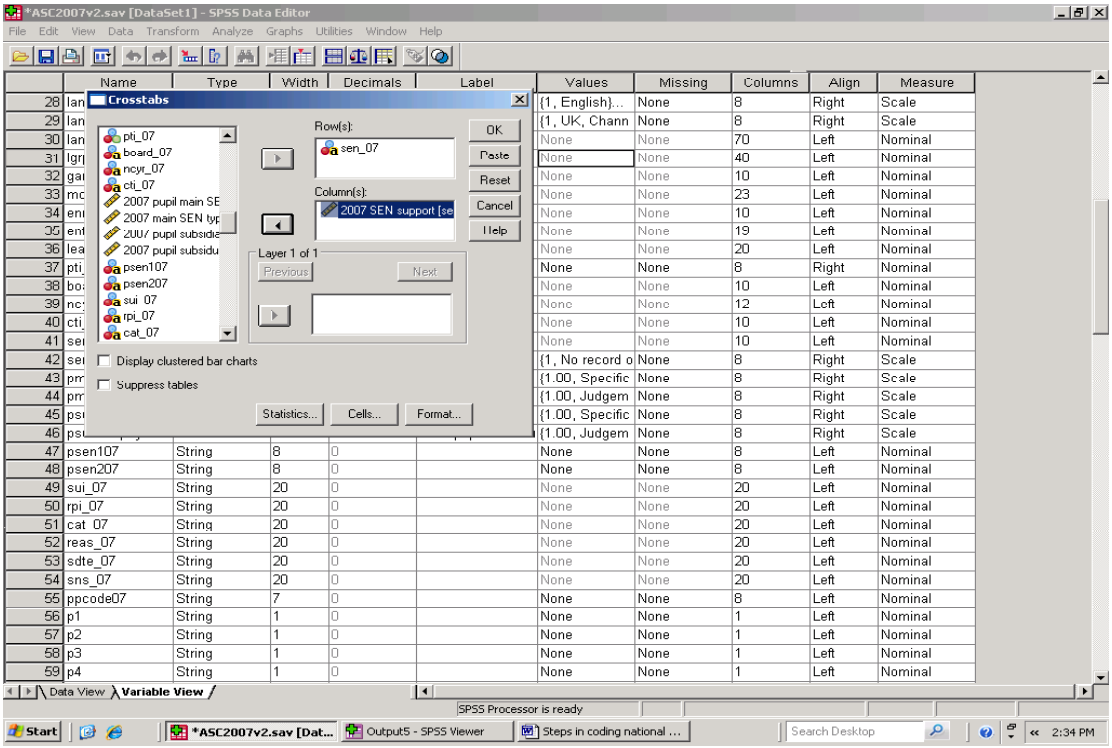


Figure 23 show that codes in the newly created variable are correctly aligned with the source text codes. All appears well, and we may be inclined to move on to whatever the next step might happen to be. However, a number of deliberate mistakes have been made.

The first involves using the 'Recode into Different Variables' procedure to recode missing (blank) values in a string variable as a numeric code in a numeric variable. That procedure does not work in SPSS, and it has been included here to illustrate that point.

(Perhaps) unsurprisingly, the grand total shown in Figure 23 is lower than the grand total of cases in the 2007 file shown in Section 5. This is the second deliberate mistake, and it is made to illustrate the point that reference to a frequency table for sen_07, which would have included cases with missing (blank) values, would have been a very useful part of this quality check.

The third deliberate error involves the new variable name and label. Section 5 introduced the suggestion that the prefix 'p' would help distinguish a variable dealing with pupil characteristics from those with the prefix 's', which would refer to a characteristic of the school attended. Neither prefix has been used in the worked example, and '2007 SEN support', which is the label give to the new numeric variable and shown in Figure 22, may not be of much help unless the Team is made up of SEN specialists. There is a limit to how much explanation can be included within such a label, but it ought to be possible to improve on the label used, and it certainly is possible to give the new variable the prefix 'p'.

Figure 23. sen_07 * 2007 SEN support Crosstabulation
Count

		2007 SEN support				
		No record of special provision	School action	School action plus	Statement of SEN	Total
sen_07	A	0	861405	0	0	861405
	N	6073835	0	0	0	6073835
	P	0	0	418873	0	418873
	S	0	0	0	221604	221604
Total		6073835	861405	418873	221604	7575717

The 'Recode into Same Variables' procedure will replace missing values in 2007 SEN support' with the number 1. Select 'Transform' from the SPSS main menu, followed by 'Recode' and 'Into Same Variables…' You will be shown a 'Recode into Same Variables' dialogue box, with a list of variables on the left. Select the new numeric SEN variable from that list and transfer it to the pane immediately to the right. When that has been done, click on the 'Old and New Values…' button, which you will see in the lower part of the dialogue box. A new dialogue box headed 'Recode into Same Variables: Old and New Values' will be shown in the screen and, apart from that title, it is identical to one illustrated in Figure 21. Select the 'System or user missing' radio button on the left of the dialogue box, and key the number 1 into the New Value section, then click on the 'Add' button followed by the 'Continue' button. This will return you to the 'Recode into Same Variables' window where you can select 'OK', and SPSS will now replace missing values in the new numeric SEN variable with 1.

9. Using a 'Compute if' statement with an added conditional 'or' to create a numeric equivalent of a string variable

Children either have a record of entitlement to free school meals or they do not, and we have seen how a string version of the FSM record can be converted into a numeric form, and given appropriate value labels to add meaning for those who read output.

In Section 8, we saw how a new numeric equivalent of an existing string variable with more than two values can be created and labelled. Section 8 also established that the 'Recode into Different Variables' procedure does not recode blank missing values in a source string variable as a number in a new numeric equivalent variable. A new conditional statement can be useful in these circumstances. The record of whether or not a pupil is on roll in a nursery class is held in the string variable *cti_YY*. These have two legitimate codes.

'N'='On roll in nursery class' and
'O'= 'on roll in other class'

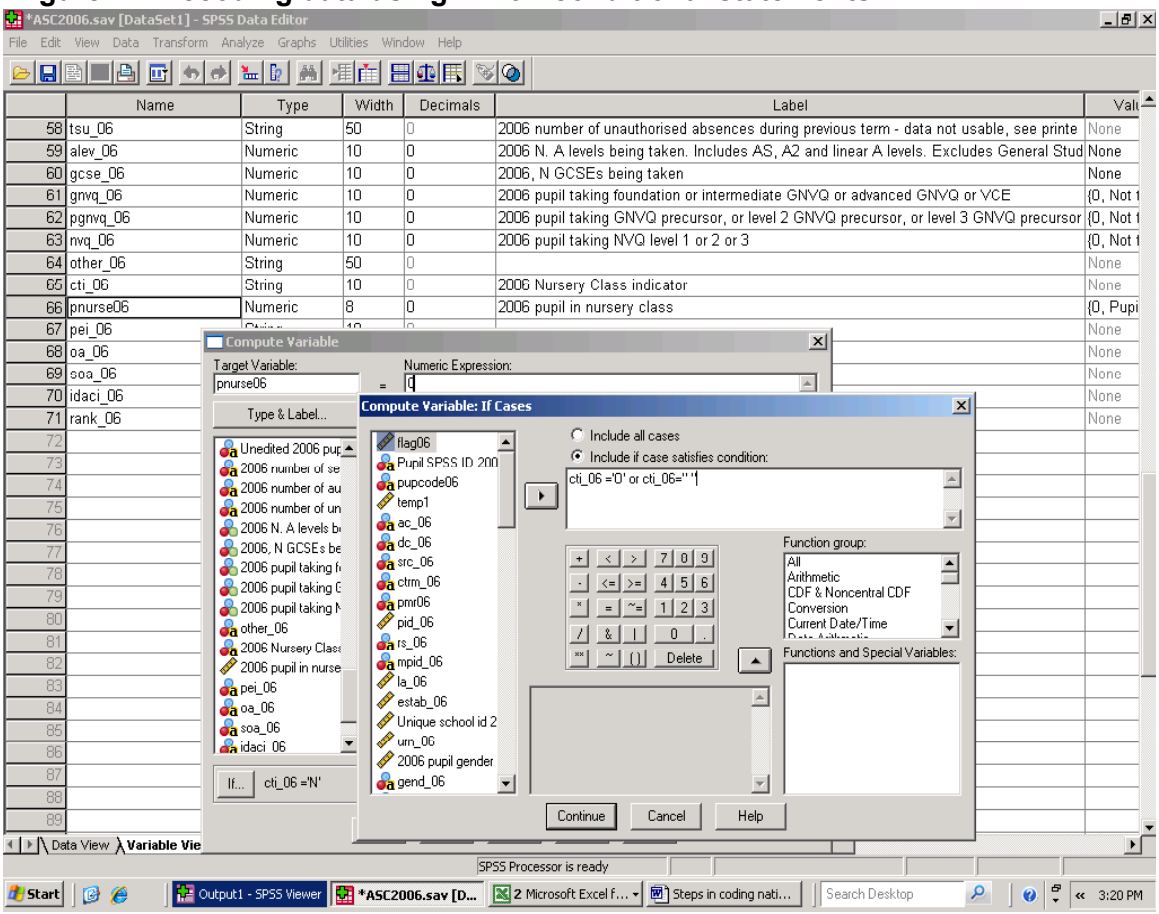
We could create a numeric equivalent of these using the 'Recode into a Different Variable' procedure. However there is, once more, a problem with missing values.

Assume that the data are for January 2006, and create a numeric variable 'pnurse06'. Locate any new variable taking whatever project is being worked on into account as a whole. In this instance the variable is given the Label '*Pupil recorded as on roll in a nursery class in 2006'*, with the Values 0=Not recorded as being on roll in a nursery class and 1=Recorded as being on roll in a nursery class.

Following Figure 24 below, compute pnurse06=0 if cti_06 if cti_06='O' or cti_06=" ". SPSS reads a space between two double quotes as meaning a blank space, and the statement reads 'give the numeric variable for nursery class roll status the value 0 if the source record contains the capital letter O or if the record is blank'.

You are likely to find this, and the facility for extending the scope of 'If' statements in SPSS, will both give a considerable degree of flexibility in future work. Other conditional statements include '&', which is referred to in Figure 59 in Section 18.

Figure 24. Recoding data using 'if' 'or' conditional statements



10. Working with string data. Upcase, Ltrim, substrings, concat, Recoding into the Same Variables and moving variables within datasets

The name 'SPSS' points to its role as a statistics package. However, the need can also arise for SPSS to organise text and alphanumeric data. (In SPSS-speak both types of data are string variables.) This Section provides introductory information on disassembling and re-assembling string variables, and takes work with postcodes for purposes of illustration. Postcodes provide a crucial link in work with pupil level data to add information from other files to the core SC file. For that to be possible, postcodes have to have the same format in all the files involved. In practice there are at least three different formats involved, and the procedures referred to in this Section can be used to standardise them in one format.

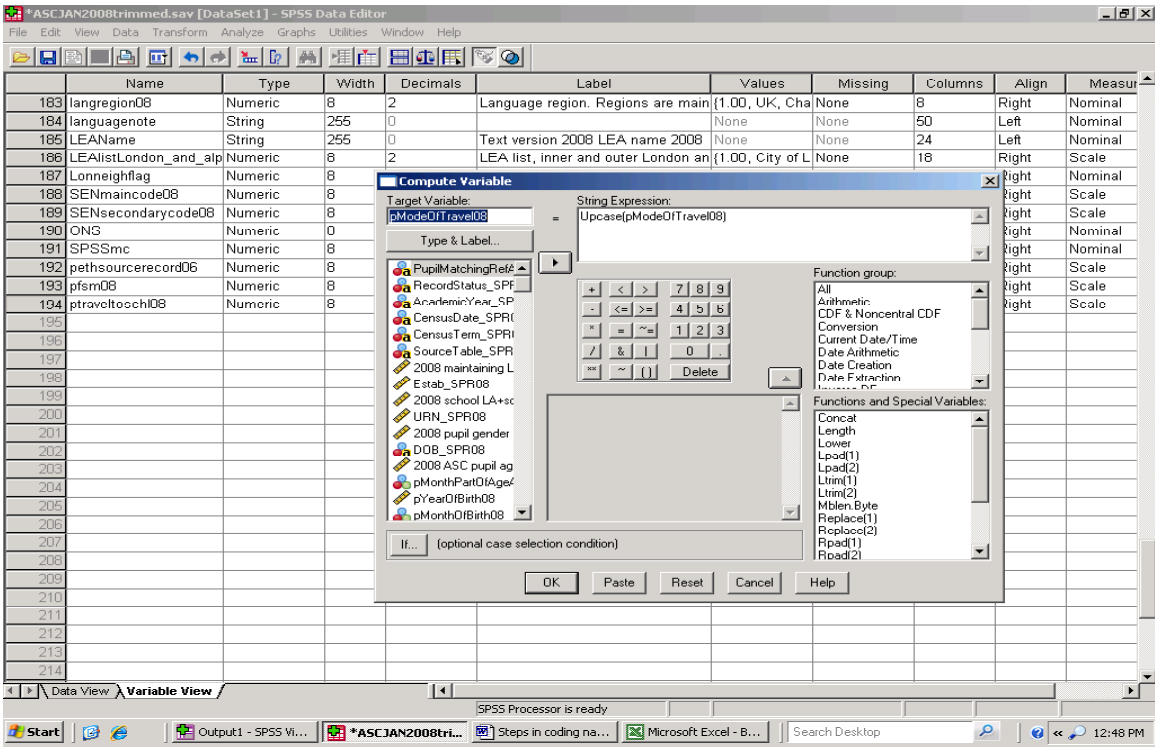
Before that, we turn to two SPSS procedures that are useful in standardising text more generally.

Figure 25 shows a variable containing codes for pupil mode of travel to school. There is an official list of string codes, all of which are in upper case. However, a frequency table shows that some schools have provided returns using a mixture upper and lower case, and in some instances entirely in lower case. The instruction

Compute
pModeOfTravel08=Upcase(PmodeofTravel08)

puts all mode of transport records in upper case. This avoids SPSS 'seeing' more modes of transport than there actually are (each separate format will otherwise be seen by SPSS as a separate mode of transport). It can also be useful for presentation reasons or, when records are to be linked to information in another file. The latter is explained in more detail in Sections 12 to 16.

Figure 25. Standardising text in upper case



A further way of standardising records involves removing leading spaces. SPSS can centre text within a variable/field, or those entering data can inadvertently press the space bar before entering a record. The record ' B' for boy is not the same, and will not be read as 'B' (notice where ' ' have been placed). To avoid disrupting analysis, unless leading spaces are part of a legitimate code, remove them.

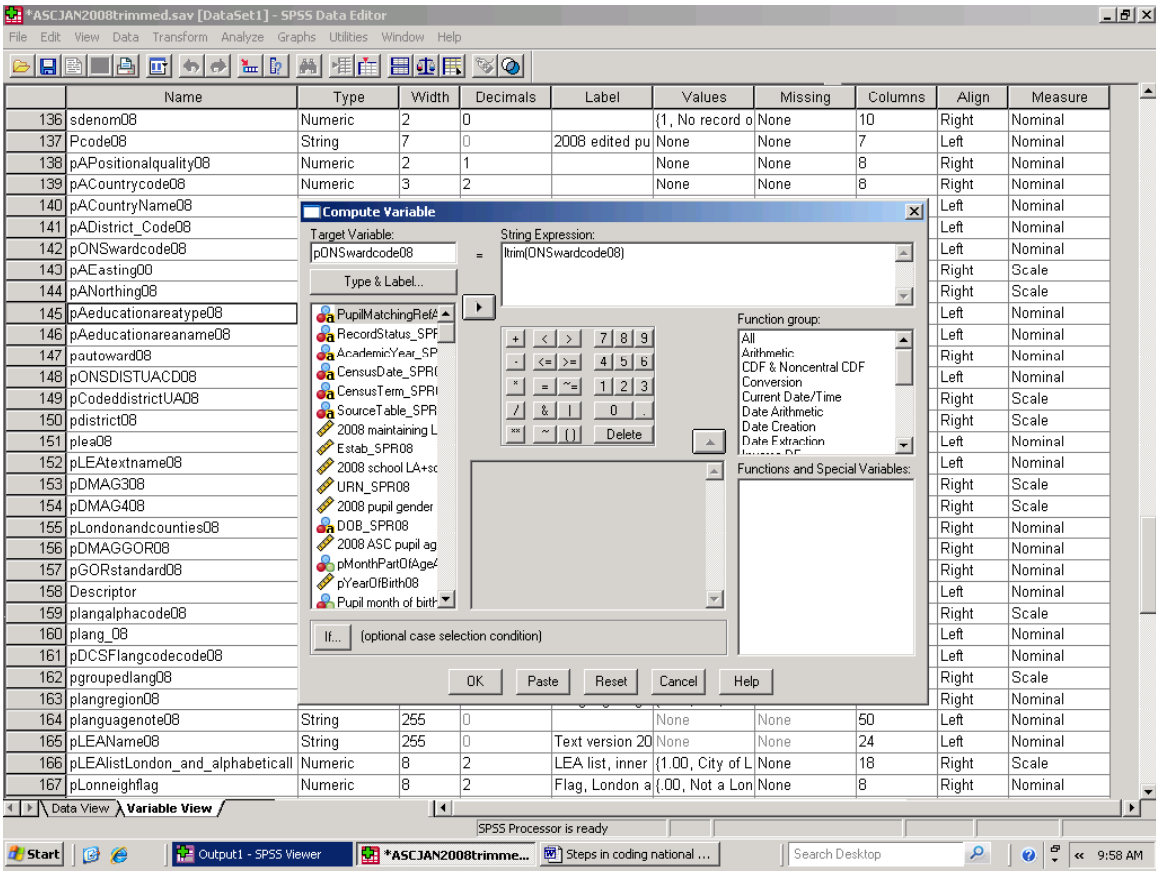
Additionally, removing those spaces can, again, be useful where records are to be linked to information in another file. Figure 26 illustrates how leading spaces are removed from a variable - in this instance the string variable ONSwardcode08. The command to achieve this is

Compute
ONSwardcode08=ltrim(ONSwardcode08)

The ltrim procedure is designed exclusively for string variables. If the 'Compute Variable' dialogue box has the heading 'Numeric Expression', click on (select) the 'Type & Label' box in the upper left hand part of the 'Compute Variable' window and select the 'String' button *before* type information into what is shown in

Figure 26 as the String Expression window. Once the information is entered, click the 'OK' button in the 'Compute Variable' dialogue box.

Figure 26. Standardising text records by removing leading spaces



The substring facility is also useful in an SPSS command to copy particular parts of a string to another variable.

Figure 27 shows the instruction for copying the first four characters of a ward code to a new variable. The command requires SPSS to deal with a part of a string, that is, with a substring (substr) In English, the command in the dialogue box is

The new variable is to be made up of characters in the ward code, starting with the first character (1) and taking the first four (4) characters for each case

Postcodes can and do have different formats, depending on the dataset in which they are held. DMAG Education works to a standardised seven-character string with the first part of the postcode left justified and the second part right justified. DCSF files use an eight-character string format for postcodes, with a single space between the first and second part of the code. CACI PayCheck files use another format.

However, if each part of a postcode in the DCSF format is shown as 1 where there is a character, and as 0 where there is a space, there are only three 'legitimate' DCSF-type UK postcode types; anything else is a miscode or involves missing data. These are

- type 1 = 11011100
- type 2 = 11101110
- type 3 = 11110111

Figure 28 is a simplified frequency table showing the number of different pupil home postcode types in the DCSF source SC file for 2006.

This binary representation of postcodes can be used to identify the format of postcodes in totally new files. It can also be used in a process that begins with disassembling 'DCSF-type' eight character postcodes and ends by recombining them in the DMAG Education seven-character format. By extension, the steps involved can be used in similar work with other codes.

Figure 27. Creating a new variable from a part of a string variable

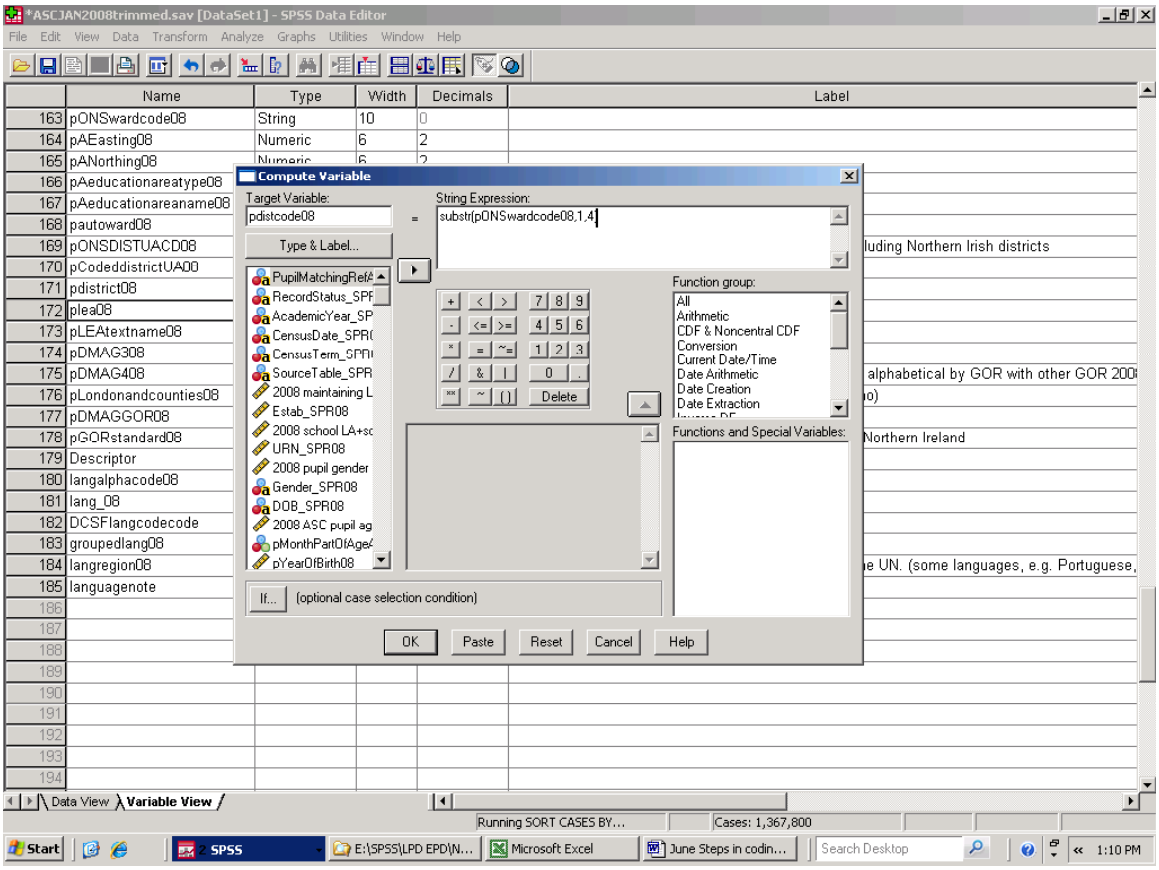


Figure 28. Simplified Frequency Table. Distribution of DCSF postcode formats, 2006

Variable		
concat	Format	Frequency
Valid	11011100	46118
	11101110	865854
	11110111	846772
	Total	1758744

Source: 2006 English Pupil Dataset

The SPSS substring facility plays a part in this, as does a facility called 'concat' facility. 'Concat' refers to concatenate, meaning to connect or link a series or chain. It can be used to recombine parts of a string variable, which have previously been separated.

The basic objective in this Section is to illustrate how: to identify which of the three types of postcode each postcode is; to disassemble an eight character postcode into its eight constituent characters, and; to reassemble them according to type in a seven-character format.

Create eight new numeric variables, c1 to c8, each of one space, and with no decimal place. The 'Recode into the Same Variables' can be used to give each of the eight new variables *with missing (blank) values* the value 1. This is a comparatively speedy operation, because all eight variables can be recoded in the same way at the same time. Key steps in achieving this are shown in the Figures 29 to 31. Select 'Transform' from the main SPSS menu, followed by 'Recode, and then 'Into the Same Variables'. In the 'Recode in Same Variables' dialogue box, select the variables to be recoded (c1-c8), and transfer them to the 'Variables' window shown in Figure 31.

Now select the 'Old and New Variables' button below the variables window. You will be shown a 'Recode into Same Variables: Old and New Values' section. Select the 'System or user-missing' (values) button in the upper left section of that window and key the number 1 into the 'New Value' section in the upper right part of the window. Click the 'Add' button on the right of the window, and SPSS will now move your instruction to the 'Old-New Section of the dialogue box. Click on the 'Continue' button, and you will be returned to the 'Recode into Same Variables' dialogue box. Click 'OK' and SPSS will now replace all missing values with the number 1. Since there is nothing other than missing values in each of the eight variables, all cases in all eight variables c1 to c8 will now be given the value 1.

Figure 29. Step 1 of Recode Into the Same Variables

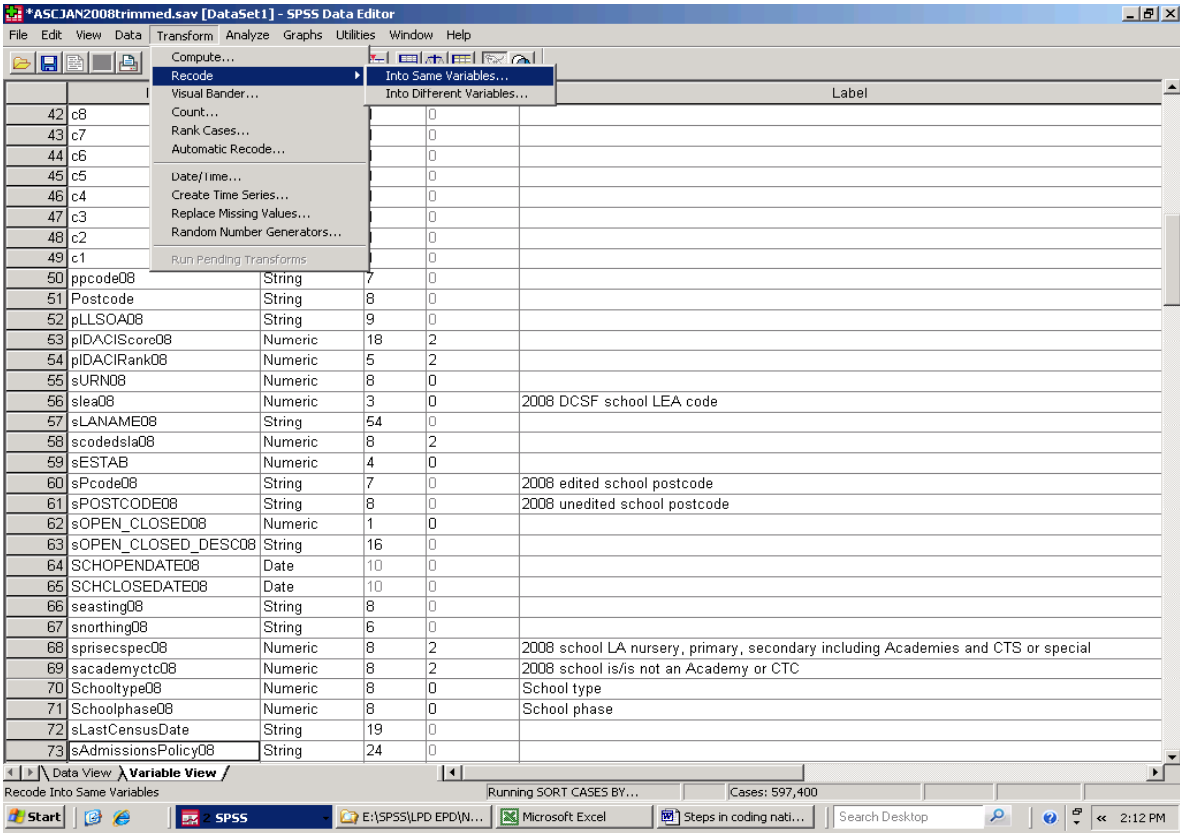


Figure 30. Step 2 of Recode Into the Same Variables

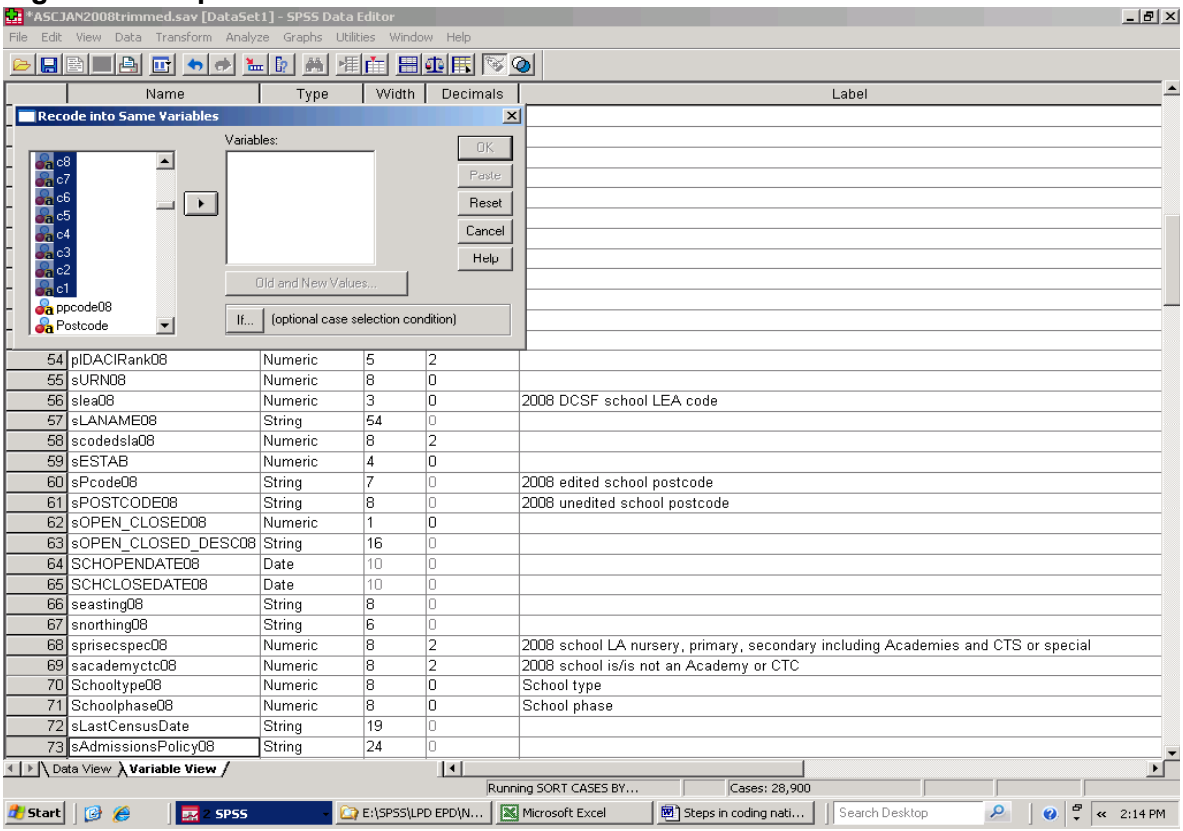
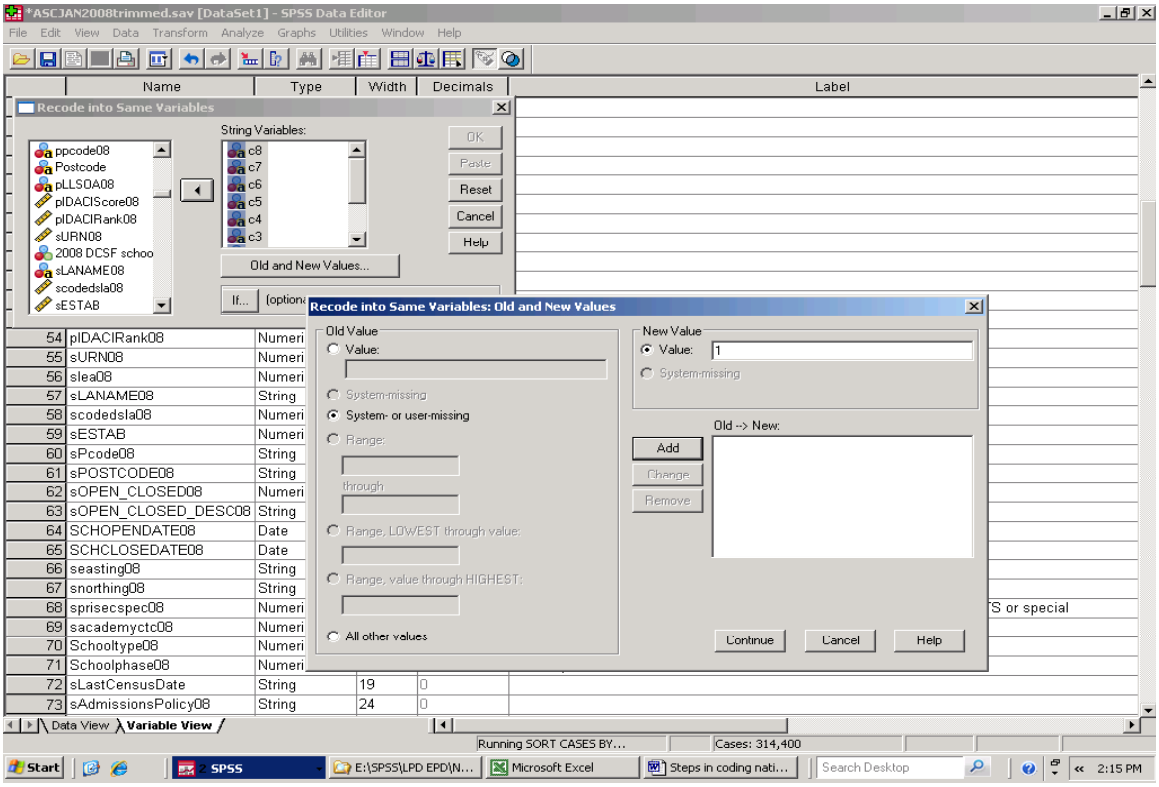


Figure 31. Step 3 of Recode Into the Same Variables



The next step is to recode each of the variables c1 to c8 as 0 (zero) to register where a space occurs in eight-character version of the postcode. This requires conditional 'Compute if' statements as described in Section 8. The first three commands are shown below.

1. Compute c=0
   if substring(uneditedpostcode,1,1)=" "
2. Compute c=0 if
   substring(uneditedpostcode,2,1)=" "
3. Compute c=0
   if substring(uneditedpostcode,3,1)=" "

In plain English the first command is saying, 'look at the first character in the postcode variable. If this is blank (SPSS reads two double quotes with one space in between, " ", as a blank space) then c1 equals 0 (zero). The second command is saying 'look at the second character in the postcode variable. If this is blank, then compute c2 equals 0'. If the command had read

Compute c=0 if substring(uneditedpostcode,2,2)=" "

SPSS would have replaced c2 with zero if the second and third characters in the postcode were blank. Put another way, the command asks SPSS to look at two characters, starting at the second character in the postcode variable. The substring command can deal with one character at a time, but it can also look at more than one character.

Complete this part of the process so that c4 is changed to zero if the fourth character in the postcode string is blank, and so on up to c8.

SPSS keeps a record of the instructions given (of which more in Section 25). The lines below are the instructions above as logged by SPSS.

```
IF (SUBSTR(Postcode,1,1)=" ") c1 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,2,1)=" ") c2 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,3,1)=" ") c3 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,4,1)=" ") c4 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,5,1)=" ") c5 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,6,1)=" ") c6 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,7,1)=" ") c7 = 0 .
EXECUTE .
IF (SUBSTR(Postcode,8,1)=" ") c8 = 0 .
EXECUTE .
```

You now have eight numeric binary variables showing where characters appear in the DCSF source postcodes. Create a new variable 'concat' and

```
COMPUTE concat =
concat(c1,c2,c3,c4,c5,c6,c7,c8)
```

This brings together the values of c1 to c8 in a single variable. To check the result, run a frequency table for the concat variable. Assuming postcodes are in the eight-character DCSF

format, the majority, and possibly all postcode records will be in one of the three legitimate DCSF-type formats shown in Figure 28, that is

11001110 (type 1 postcodes)
11101110 (type 2 postcodes)
11110111 (type 3 postcodes)

All being well, you can now create a numeric variable called 'Type' using the Compute and IF procedures.

Compute Type=1 if concat='11011100'
Compute Type=2 if concat='11101110'
Compute Type=3 if concat='11110111'

The next step is to create 7 string variables P1 to P7, each one character wide. These are to take information from the source postcode record, in the position needed in the standardised seven character version the postcodes. The seven variables can then be combined using the concat facility to give a reformatted, seven character, postcode. The issue is how to decide what part of the source postcode goes in which of the seven variables P1 to P7.

One key factor is that in the seven character version of postcodes the first part, for example SE1 in SE1 2AA, will be left justified (there will be no leading spaces). Another key factor is that and the second part of the postcode, in this example 2AA, will be right justified (there will be no trailing spaces).

Type 1 unedited postcodes in DCSF format are therefore converted from a 1101110 format to a 1100111 format.
DCSF *Type 2* unedited postcodes are converted from a 11101110 format to a 1110111 format.
DCSF *Type 3* unedited postcodes are converted from a 11110111 format to a 1111111 format.

Allocating the first two characters is straight-forward since all legitimate English postcodes begin with two characters. In this instance the next two steps are simply to

Compute P1 = substr(unedited postcode,1,1)
Compute P2 = substr(unedited postcode,2,1)

SPSS has now given P1 the first character of the unedited postcode and P2 the second character of the unedited postcode. However, after that point, allocating information to the remaining 5 'P' variables will be determined by the type of postcode in question

In their edited form, the third and fourth character spaces of Type 1 post codes are to be left blank. The third and fourth characters will be made up of P3 and P4. These can be left as they are (empty) and the next three steps with Type 1 postcodes are

Compute P5 = substr(unedited postcode,4,1) if Type = 1
Compute P6 = substr(unedited postcode,5,1) if Type = 1
Compute P7 = substr(unedited postcode,6,1) if Type = 1

Type 2 postcodes have a 11101110 format in the DCSF source data. In their case P4 needs to be left blank and P3, P5, P6 and P7 calculated as follows

Compute P3 = substr(unedited postcode,3,1) if type = 2
Compute P5 = substr(unedited postcode,5,1) if type = 2
Compute P6 = substr(unedited postcode,6,1) if type = 2
Compute P7 = substr(unedited postcode,7,1) if type = 2

The next steps for Type 3 unedited postcodes are

Compute P3 = substr(unedited postcode,3,1) if type = 3
Compute P4 = substr(unedited postcode,4,1) if type = 3
Compute P5 = substr(unedited postcode,6,1) if type = 3
Compute P6 = substr(unedited postcode,7,1) if type = 3
Compute P7 = substr(unedited postcode,8,1) if type = 3

Concatenating variables P1 to P7 will now provide an edited standardized seven character postcode, Create a seven character 'edited pupil home postcode' variable 'ppcode', with a year designator, e.g. ppcode07 (for 2007) and compute

ppcode(YY) = concat(P1,P2,P3,P4,P5,P6,P7)

Delete the temporary c1 to c8 and P1 to P7 variables and save the file.

You can insert these new variables exactly where you want them to be within the dataset, and then use the procedures outlined earlier, and then use the Compute procedure. Alternatively, the user can skip the 'Edit/'Insert Variable' steps, and use the Compute procedures straight away. SPSS will then simply add new variables to the end of the dataset.

In this instance, it is best if new variables referred to in this Section are kept together, since this will allows for a visual check on data as work proceeds. Equally importantly, disorganising a dataset by placing new variables in an unplanned

manner just makes it increasingly difficult to locate the variables needed in analyses.

For those who reach a point where housekeeping is needed to reorganise a dataset, new copies of existing variables can be created, in the position they are needed, following either of two routes. Both of these begin with using the Edit, Insert Variable command to create an empty variable into which an untidy variable can be moved. The first route is not suitable for work with large datasets, where it will simply lock SPSS.

In the first method, and in SPSS Data View, select (click on) the name of the variable to be moved, and then select 'Edit' from the SPSS main menu followed by 'Copy'. After this, in a similar manner select the new blank variable name in Data View, and select 'Edit' from the main SPSS menu, followed by 'Paste'. The existing variable will be copied, with any value labels to its new location. (A standard Windows 'Cut' and 'Paste' would achieve much the same result, but would leave the user without the parachute that 'Copy' and 'Paste' provides. For those starting out on moving variables, 'Copy' and 'Paste' provides the safer route in the early days).

The second method is

Compute new variable=old variable.

If the variable in question has value labels, simply go to the SPSS Variable View select the Values cell of the existing variable, and then select 'Edit' from the main SPSS menu followed by 'Copy'. After this, select the Values cell of the new variable followed by selecting 'Edit' from the main SPSS menu and 'Paste'.

While work with postcodes at City Hall is mainly tied to merging datasets, the concat and substring facilities will have a wider application. That said, the Guide now turns to processes and pitfalls in merging datasets.

11. Pre-amble to merging files. Using an existing data dictionary from another file

SPSS maintains a record of the value labels created for variables. Those value labels can be viewed on a variable-by-variable basis by selecting a variable's Values cell in the SPSS Variable View window, or by using the command shown on page 13, which will show the record for all variables in the dataset. Depending on the version of SPSS being used, that record is referred to either as a Data Dictionary or as Data Properties.

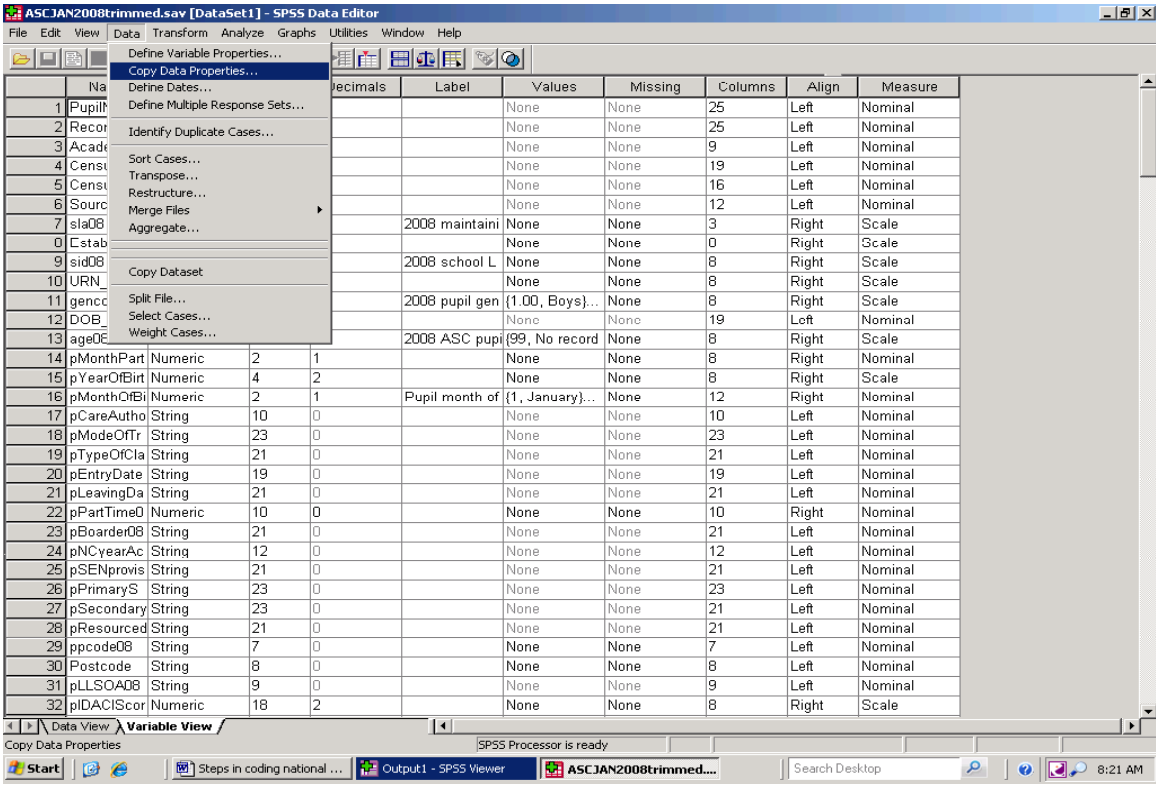
Value labels can be created quickly and easily for a variable by applying the data dictionary from another file – if the appropriate value labels have already been created in that other file, and if the variable exists in both files and has the same type of content. Value labels can be created for several variables simultaneously in this way, and this will represent a considerable saving in time in work with large files.

For the sake of simplicity, the worked example shows the application of the data dictionary for one variable in one file to its unlabelled equivalent in another file. The source data in the 2008 SC file includes the individual's month of birth as a numeric variable, with the number referring to the sequence of the month, but there are no value labels. Assume that another file contains similar information but with value labels added to give month 1 the label 'January', month 2 the label 'February' and so on.

Figures 32 to 36 show the sequence of steps that will apply the value labels from that file to the numeric data in the 2008 SC file. Open the file which is to receive value labels and, in Variable View, Select 'Data' from the main menu and then select 'Copy Data Properties' from the resulting drop-down list. This opens the first of the 'Copy Data Properties Wizard' windows, shown in Figure 33,and begins a step-by-step procedure of the sort covered in the account the file import wizard in Section on page 14.

The dialogue box shown in Figure 33 enables the user to select the file which already has the value labels needed. Select the appropriate file and then click on 'Next'.

Figure 32. Step 1 Copying Data Properties – aka applying Data Dictionaries



The 'Copy Data Properties – Step 2 of 5' window which follows enables the user to specify which variables value labels are to be copied from. The selections shown in Figures 34 and 35 below mean that the value labels will only be given to a variable with exactly the same name in the 2008 SC file. Having made the selections, click 'Next' and, the Copy Data Properties Step 3 window follows. This is where the properties of the coded and labelled variable that will applied to the coded but unlabelled variable are determined. In this instance, the assumption of that there are no value labels in the 2008 SC file, and that such labels as will be added are from the external file.

Again, click the 'Next' button. In this instance ignore the next window by selecting 'Next', and value labels will now be applied to the 2008 SC file virtually immediately. (Compared with a number of procedures in SPSS, this one is applied remarkably quickly).

Figure 33. Step 2 Copying Data Properties – aka copying Data Dictionaries

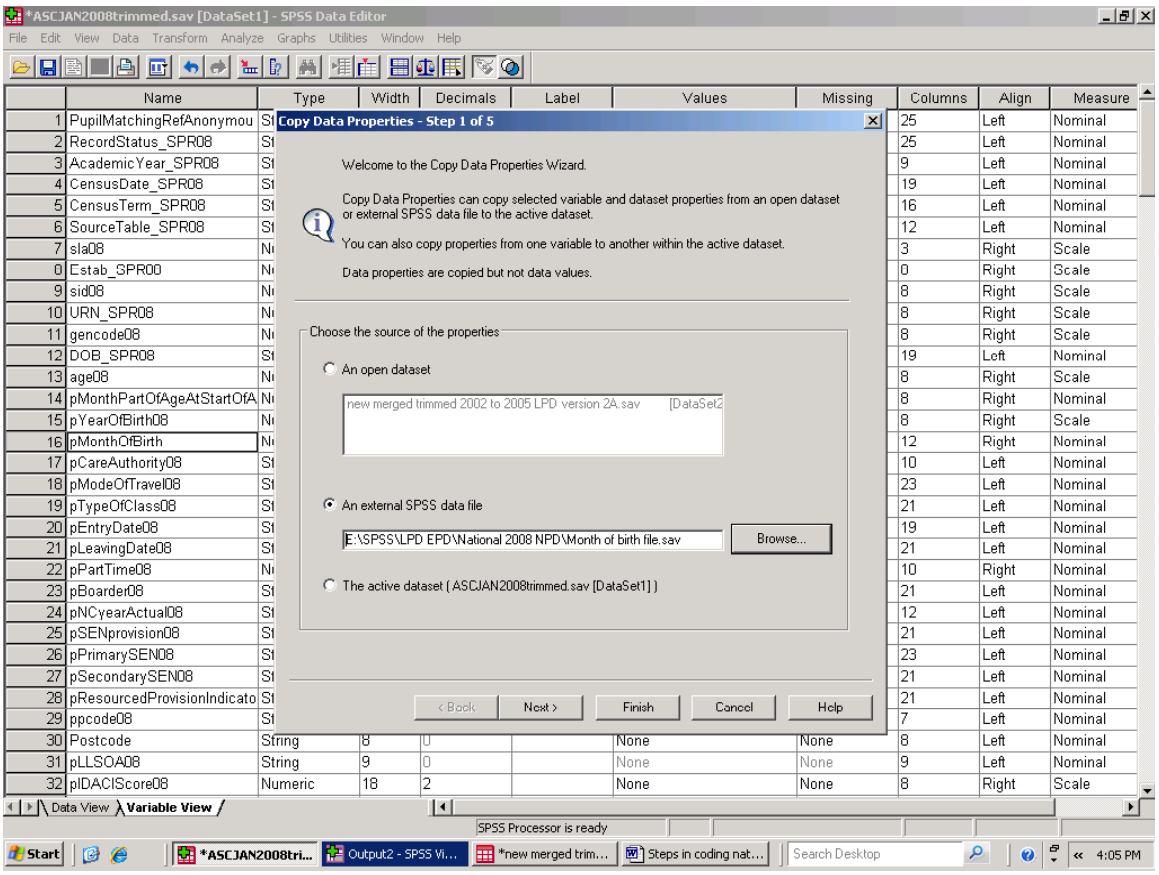


Figure 34. Step 3 Copying Data Properties – aka copying Data Dictionaries

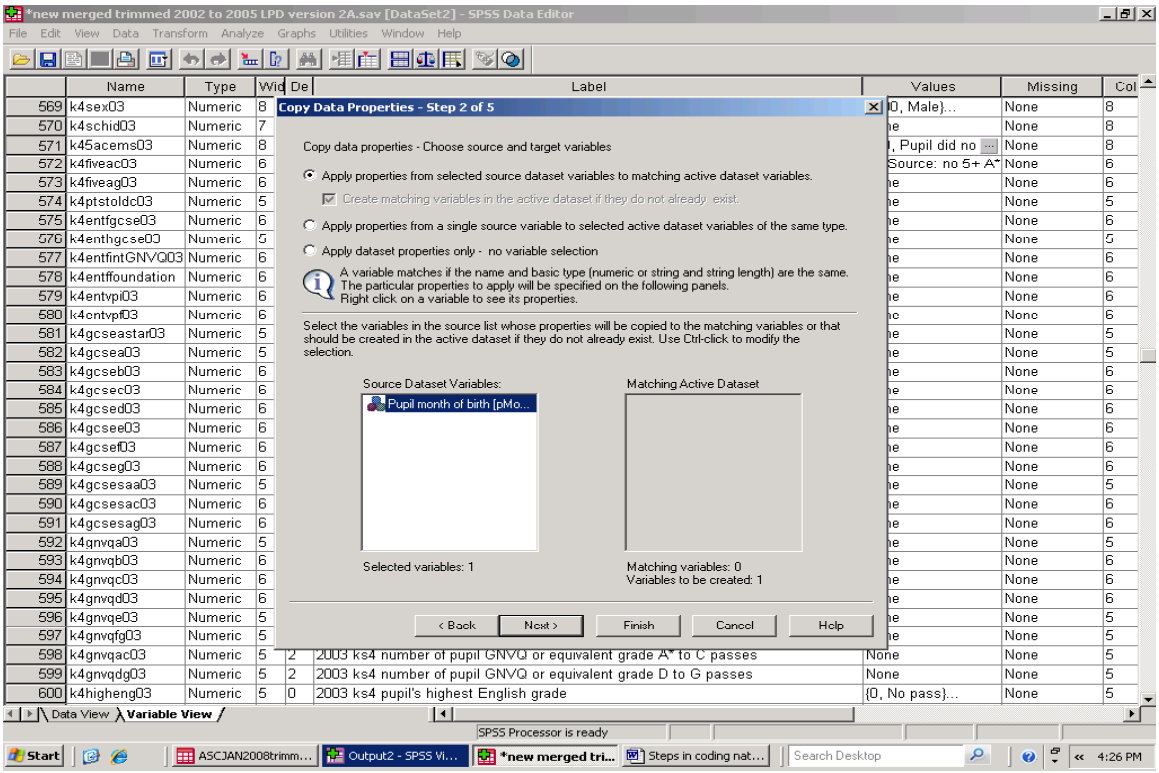


Figure 35. Step 4 Copying Data Properties – aka copying Data Dictionaries

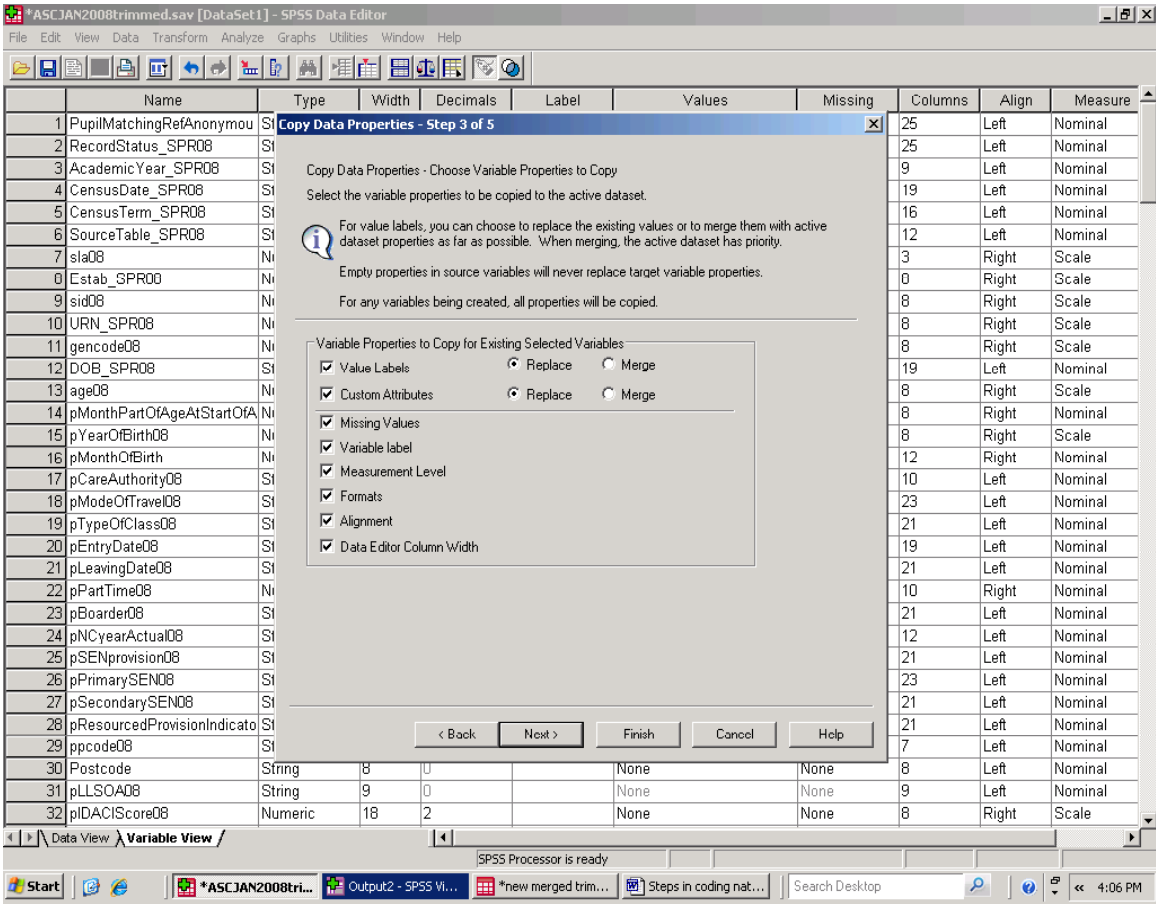
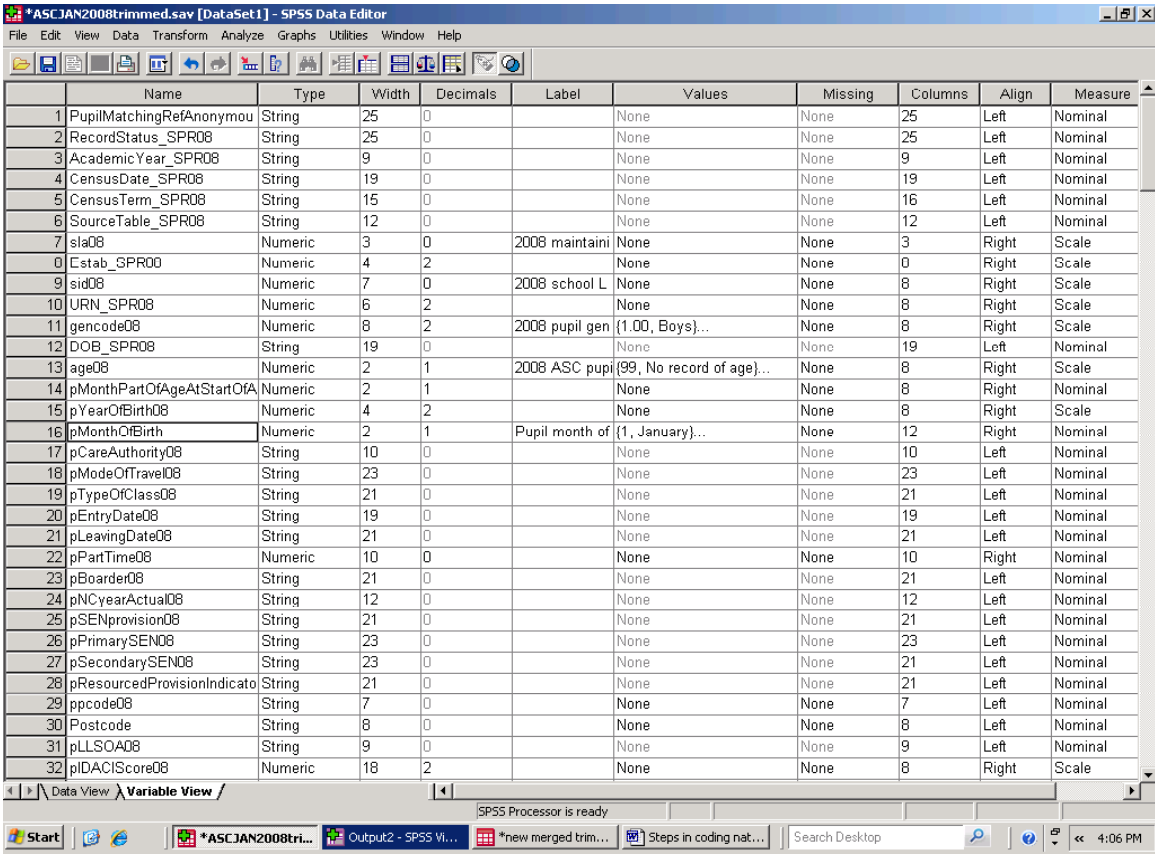


Figure 36. Data properties are copied – value labels are established



A new and labelled numeric equivalent of an existing string variable cannot be created in this way by using the Copy Data Properties procedure and, what can be much the same thing, data cannot be added from one file to another following this procedure. However, the Copy Data Properties facility will be a real asset where a dataset has a large number of numeric variables, and where the codes for these exist already in an external file.

However, useful as that facility can be, there is a more general point. There are circumstances in which the potential of one dataset can be enhanced by adding information from another dataset. Those working with data will need to understand the SPSS procedures involved, and they will also need to keep a weather eye on what existing, new and developing datasets 'out there' may have to offer in the future.

12. Merging datasets. The order of events in using external lookup tables

The previous section provided an example of adding information (value labels) from a numeric variable in one dataset to a numeric variable in another dataset. However, a wide range of other variables in the NPD are string variables, with there own codes. For example, the 2007 English National Pupil Dataset contains a string 'pupil home language' variable. Each of more than 340 languages has its own recognised string code. The codes are not particularly meaningful on their own, and there are simply too many for the mere mortal to memorise. SPSS string variables are more demanding of computing capacity and, in any event, long string variables cannot be used in some SPSS procedures such as the Tables procedure.

The solution is to create numeric equivalents of the string codes, and to give these value labels, However, where a variable takes on 340 different values, using the 'Recode into Different Variables' facility described in section 8 will be time-consuming, as well as exceedingly boring. The Recode facility will, in any event, not work as a single exercise when applied to a large number of values.

As an alternative, the Merge File facility can be used to add variables from another dataset where those string codes have already been given a numeric equivalent, and where those numbers have each been given a value label. That 'other' dataset is, in essence, a lookup table with each language code appearing once and only once. The lookup table may well still need to be created, but it can make sense to do that if the lookup table can be used on several occasions with, for example, data from different years.

For most purposes, using the Merge File facility to add information from an external lookup file requires

- a 'key' variable to link datasets,
- which is present in the lookup table and in the file lacking value labels, and
- has exactly the same name, character and level of measurement in the both files
- and which has been sorted in the same, usually ascending order, in both files.
- Finally, as a general rule the lookup dataset cannot contain duplicate codes. (How to identify duplicate records and remove records is described in sections 14 and 6 respectively.)

For those drawing on short text codes in data from relational databases, the Merge File facility can be particularly important. It is that facility which is used to add information on assessments from the pre-school years to the end of schooling to the 'PLASC' file. If those data were not added, the purely educational value of 'PLASC' would be vanishingly small.

Additionally, in City Hall pupil level datasets are linked on postcode to other datasets that are not part of the NPD. That type of file merge has added information on pupil home area and school location, including location (ward, borough, region, eastings and northings). In the same way, information on equivalised income at a small area level has been added to pupil records in a number of SC files from different years. Each school has a unique identifying code, and this has been used to add to pupil records a wide range of information on the characteristics of the school each pupil attends.

The first type of work can be described in general terms as adding labelled numeric equivalents to string variables that already exist within the NPD. In general terms the second body of work can be described as adding information to pupil records from datasets beyond the NPD. However, the procedures involved in both are much the same though, as Section 16 shows, particular datasets can raise particular issues. Indeed, datasets which exist separately from the NPD may themselves need to have string variables converted to numeric, coded and labelled, equivalents.

The procedures described in the Section can, then, be useful in a variety of situations. However, caution is needed if the datasets involved are large. At the least, running the sorting and merging procedures will be time-consuming, and work with pupil level data in City Hall has been left to run overnight on a number of occasions. Access to additional computer might be considered so that, resources permitting, other work can proceed.

Additionally, while the aim in merging files is to add variables required in a research project, it also follows that this increases file size, and further slows down data processing. We will in Section 17, (some) SPSS procedures can fail when applied to very large datasets. In extreme cases, a dataset can be corrupted if work stretches SPSS a computer's capacity. (which is where that backup copy becomes useful). At a minimum, check how large the files are you wish to work with, and how much disc space is available to you (see page 55). If in doubt, seek advice *before* adding large numbers of records to a dataset.

As a general rule, use frequency tables to check (review) the completeness of variables you plan

to work with. If there is choice of reviewing the quality and completeness of the same variable/s in more than one dataset, as a general rule, check data in the smaller dataset. Frequency tables are run more quickly on smaller than on larger datasets. If the aim is to add a large number of variables from an external dataset with comparatively few cases, check that new data in the external dataset for completeness *before* adding them to your main dataset. If a few variables held in a large dataset are to be added to a small dataset, consider checking them once the two files have been merged.

File mergers add variables to the end of the list of existing variables in the working file (or, for those thinking in spreadsheet terms, at right hand end of data in a worksheet). Where single coded and labelled numeric variable is being added to the working file, the user will wish to consider whether the end of the dataset is the best place for it. SC datasets provided by DCSF put variables in groups, and this is likely to be the case with other datasets. For example, the variables for gender, ethnicity, free school meal entitlement, care status, language and SEN are in close proximity as some of the first variables in the SC file. You may well wish to maintain variable groupings where variables are used together or in sequence in analyses of – in this instance - educational attainment. Adding variables randomly, and placing them randomly within datasets, just makes finding the variables needed more difficult. Housekeeping work with datasets is referred to in pages 39 and 40. Planning the time needed for file mergers can usefully include planning for any associated housekeeping.

In principle, new numeric variables can be created to replace all their string equivalents in the sequence in which the string variables appear in the source SC file. SPSS will add each variable in turn to the end of the list of variables. On the other hand, unless the project in hand has a very narrow focus and a small number of variables, the realities of work will not allow all possible coding to be done as a single exercise. Consider the coded dataset listed in Appendix 1. The need for elegant simplicity in dataset structure has to be traded against research priorities, and these can and will change over time. Research priorities as much as data management drove the structure of the file illustrated in the Appendix. This does not mean that a chaotic dataset is OK; a dataset needs some coherence if users are to be able to work with it.

Label variables as soon as possible after a file merge has taken place. If the variable name needs to be altered, that should also happen as soon as possible after a file merge. 'Old' string versions of variables can also be deleted after a file merge (though not before checks are run on their new numeric equivalent to ensure that a file merger has proceeded as intended).

The variables showing a pupil's type of special educational need (SEN) provide a short working example illustrating the creation of lookup tables and how to use the Merge File procedure. Main and subsidiary 'official' DCSF SEN type codes are shown in Figure 37, with the meaning of the code on the right. A 'lookup table' can be created in a new blank SPSS dataset which summarises the list above, and which can be merged with the relevant SC file.

Five conditions need to be met.

(a) In this approach, the lookup file and the SC file will be matched and merged on a single key variable found in both files. The key variable will have exactly the same properties, e.g. name, character (string or numeric) and width in both files.

(b) The two files need to be sorted on the same key variable in ascending order before the files can be merged.

(c) The lookup file needs to contain the 'official' text codes which DCSF requires schools to use in the year when the SC data were collected. (Also see page 34).

(d) There can be no duplicates in the lookup table's key variable

(e) The lookup table also needs a separate numeric variable for each code, to which a label has been attached. For example 'ASD' could be accompanied by the numeric code 1, with the numeric code 1 being given the label 'Autistic Spectrum Disorder'.

Figure 37. SEN codes and their meaning

psen107	Meaning of code
ASD	Autistic spectrum disorder
BESD	Behaviour, emotional and social difficulty
HI	Hearing impairment
MLD	Moderate learning difficulty
MSI	Multi-sensory impairment
OTH	Other difficulty/disability
PD	Physical disability
PMLD	Profound and multiple learning difficulty
SLCN	Speech, language and communication difficulty
SLD	Severe learning difficulty
SPLD	Specific learning difficulty
VI	Visual impairment
Variable blank	No record of SEN or missing data

The national SC file contains records for more than 7 million individual pupils, and some of the codes used in schools may be incorrect. These cannot be matched to 'official' DCSF codes, and there are several approaches to dealing with them

depending on whether there is value in being able to identify the number of miscodes in a particular year.

Where there is a point in identifying the number of miscodes, you will in any event have already run a frequency table on the (e.g.) psen107 variable in the SC file. Check that Table against the official codes. If miscodes exist, these *can* be included in the lookup file and given the same numeric code and a value label of 'miscode'. This can be particularly appropriate as a quality check on data if it is being collected in a research project for the first time. (Also, see Section 7 of the Guide on Missing Values). One option is to ignore miscodes. These will not be given a value label in the file merge; they will have a missing value. In some instances, a miscode may be unacceptable, and considerable effort may be needed to correct for miscodes and/or missing data. Pages 63 to 65 refer to a real world situation of that type.

In a step by step approach to merging files

1 create the lookup file, in this case SENtype08.sav, using the same code for pupil main SEN type that is used in SC 2008 (psen108) and making sure that the variable characteristics on this key are identical. A numeric variable, which is the equivalent of the string codes, is needed with the value labels shown on page 29. The codes can be typed in directly in SPSS Data View, with value labels added in SPSS Variable View, as shown Figure 11. The ordering of the code numbers is important, and is discussed at the end of this Section. Sort the lookup file on psen108 and save it in an appropriate place. In City Hall, the file is saved in the SEN subfolder of a Coding Systems folder.

2 Close the lookup table SENtype08.sav

3 Open the main file (SC2008 file) and sort the dataset on the link variable (psen108) in ascending order. The larger the dataset, the longer this will take. Sorting the SC on string variable can take several hours. When the sorting is complete, save the main file and leave it open.

4 Select (click on) 'Data' and 'Merge File' from the SPSS main menu, followed by 'Add Variables', as in Figure 38. Then select 'Browse' from the resulting 'Add Variables to…..' dialogue box, as illustrated in Figure 39. Locate the lookup table and left click on it its name.

Figure 38. Merging files – step 1

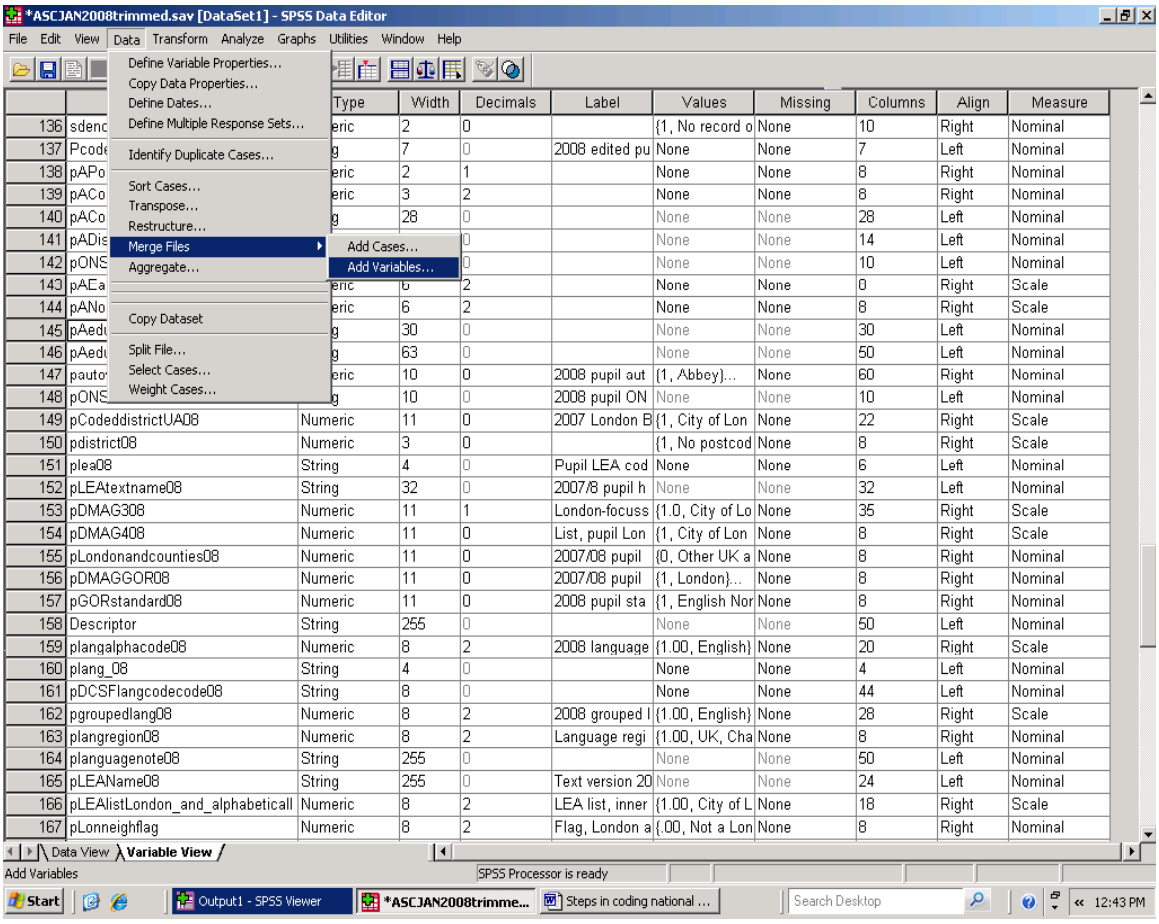
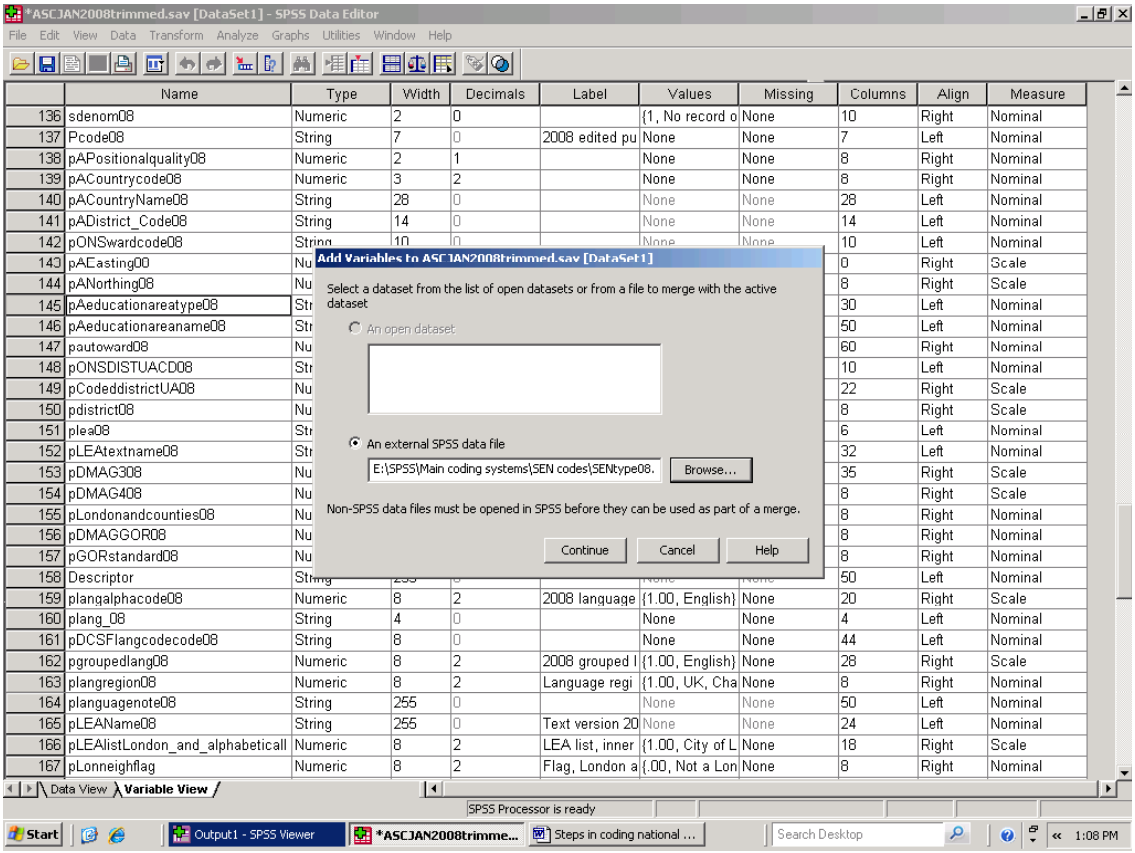
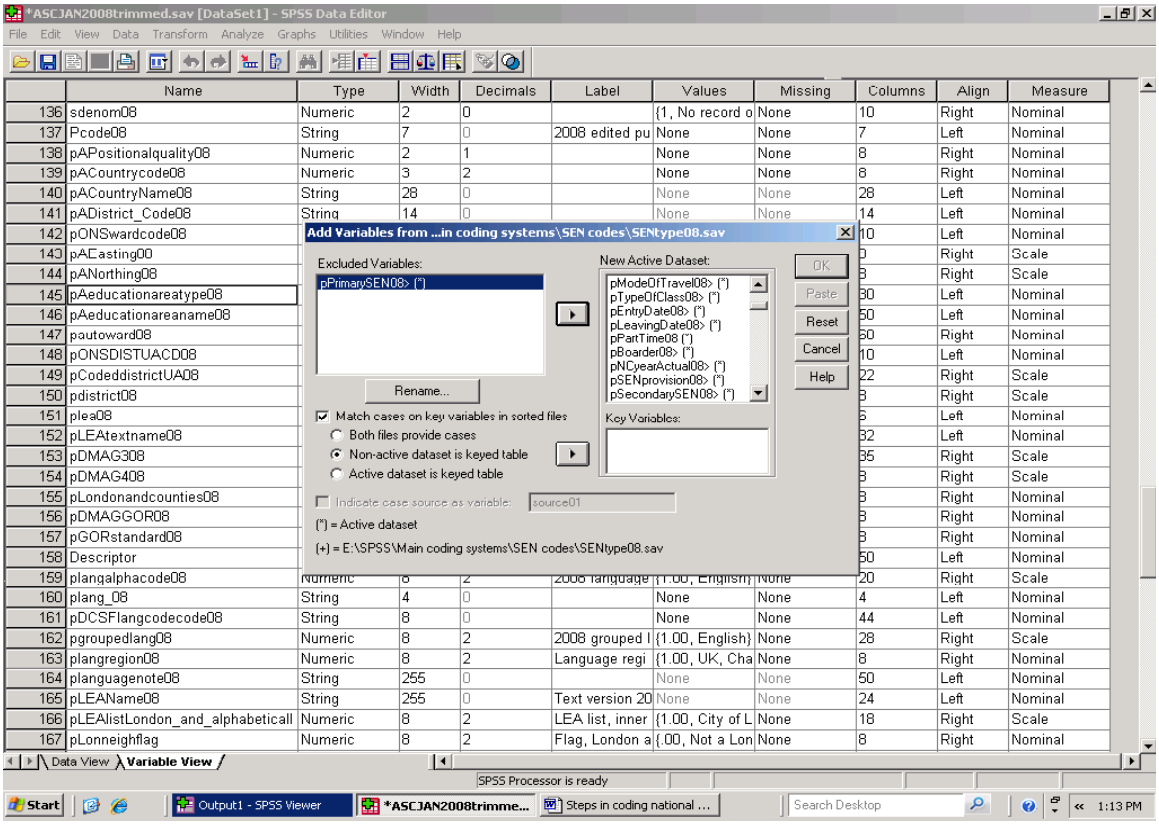


Figure 39. Selecting an external (lookup) dataset for file merger



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
136	sdenom08	Numeric	2	0		{1, No record o	None	10	Right	Nominal
137	Pcode08	String	7	0	2008 edited pu	None	None	7	Left	Nominal
138	pAPositionalquality08	Numeric	2	1		None	None	8	Right	Nominal
139	pACountrycode08	Numeric	3	2		None	None	8	Right	Nominal
140	pACountryName08	String	28	0		None	None	28	Left	Nominal
141	pADistrict_Code08	String	14	0		None	None	14	Left	Nominal
142	pONSwardcode08	String	10	0		None	None	10	Left	Nominal
143	pAEasting00	Nu						0	Right	Scale
144	pANorthing08	Nu						8	Right	Scale
145	pAeducationareatype08	Str						30	Left	Nominal
146	pAeducationareaname08	Str						50	Left	Nominal
147	pautoward08	Nu						60	Right	Nominal
148	pONSDISTUACD08	Str						10	Left	Nominal
149	pCodeddistrictUA08	Nu						22	Right	Scale
150	pdistrict08	Nu						8	Right	Scale
151	plea08	Str						6	Left	Nominal
152	pLEAtextname08	Str						32	Left	Nominal
153	pDMAG308	Nu						35	Right	Scale
154	pDMAG408	Nu						8	Right	Scale
155	pLondonandcounties08	Nu						8	Right	Nominal
156	pDMAGGOR08	Nu						8	Right	Nominal
157	pGORstandard08	Nu						8	Right	Nominal
158	Descriptor	Str						50	Left	Nominal
159	plangalphacode08	Numeric	8	2	2008 language	{1.00, English}	None	20	Right	Scale
160	plang_08	String	4	0		None	None	4	Left	Nominal
161	pDCSFlangcodecode08	String	8	0		None	None	44	Left	Nominal
162	pgroupedlang08	Numeric	8	2	2008 grouped l	{1.00, English}	None	28	Right	Scale
163	plangregion08	Numeric	8	2	Language regi	{1.00, UK, Cha	None	8	Right	Nominal
164	planguagenote08	String	255	0		None	None	50	Left	Nominal
165	pLEAName08	String	255	0	Text version 20	None	None	24	Left	Nominal
166	pLEAlistLondon_and_alphabeticall	Numeric	8	2	LEA list, inner	{1.00, City of L	None	18	Right	Scale
167	pLonneighflag	Numeric	8	2	Flag, London a	{.00, Not a Lon	None	8	Right	Nominal

Figure 40. Choosing the linking variable and identify the variables that are not to be added



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
136	sdenom08	Numeric	2	0		{1, No record o	None	10	Right	Nominal
137	Pcode08	String	7	0	2008 edited pu	None	None	7	Left	Nominal
138	pAPositionalquality08	Numeric	2	1		None	None	8	Right	Nominal
139	pACountrycode08	Numeric	3	2		None	None	8	Right	Nominal
140	pACountryName08	String	28	0		None	None	28	Left	Nominal
141	pADistrict_Code08	String	14	0		None	None	14	Left	Nominal
142	pONSwardcode08							10	Left	Nominal
143	pAEasting00							0	Right	Scale
144	pANorthing08							8	Right	Scale
145	pAeducationareatype08							30	Left	Nominal
146	pAeducationareaname08							50	Left	Nominal
147	pautoward08							60	Right	Nominal
148	pONSDISTUACD08							10	Left	Nominal
149	pCodeddistrictUA08							22	Right	Scale
150	pdistrict08							8	Right	Scale
151	plea08							6	Left	Nominal
152	pLEAtextname08							32	Left	Nominal
153	pDMAG308							35	Right	Scale
154	pDMAG408							8	Right	Scale
155	pLondonandcounties08							8	Right	Nominal
156	pDMAGGOR08							8	Right	Nominal
157	pGORstandard08							8	Right	Nominal
158	Descriptor							50	Left	Nominal
159	plangalphacode08	Numeric	8	2	2008 language	{1.00, English}	None	20	Right	Scale
160	plang_08	String	4	0		None	None	4	Left	Nominal
161	pDCSFlangcodecode08	String	8	0		None	None	44	Left	Nominal
162	pgroupedlang08	Numeric	8	2	2008 grouped l	{1.00, English}	None	28	Right	Scale
163	plangregion08	Numeric	8	2	Language regi	{1.00, UK, Cha	None	8	Right	Nominal
164	planguagenote08	String	255	0		None	None	50	Left	Nominal
165	pLEAName08	String	255	0	Text version 20	None	None	24	Left	Nominal
166	pLEAlistLondon_and_alphabeticall	Numeric	8	2	LEA list, inner	{1.00, City of L	None	18	Right	Scale
167	pLonneighflag	Numeric	8	2	Flag, London a	{.00, Not a Lon	None	8	Right	Nominal

5 An ‘Add variables from ….’ window will appear (illustrated in Figure 40) with a ‘Excluded Variables’ section listing variables which appear in both the lookup file and in SC2008. These variables will be excluded from the file merger. SPSS will not accept two variables with the same name in the same dataset. All other things being equal, these will include the key variable (pPrimarySEN08) which will be used to link the two files.

6 Select pPrimarySEN08 in the ‘Excluded Variables’ section of the dialogue box.

7 Click on the ‘Match cases by key variables’ box immediately below and then Select ‘Non-active dataset is keyed table’ immediately below.

8 Click on the arrow button to the right, which will transfer the name of the link variable name (pPrimarySEN08) to the Key variables pane.

9 The prompt will appear ‘Warning: Keyed match will fail if data are not sorted in ascending order of Key Variables’. As long as both files have been sorted in this way, click on ‘OK’

10 SPSS will prompt the user to save the SC2008 file. If it has already been saved click on ‘No’ and the file merge will begin. If the steps have not been followed either save SC2008 or cancel the file merge exercise as circumstances require.

11 psen108 will not be added to SC2008 (it is already there) but the variable containing the numeric equivalents, with their value labels, will be added to the end of the SC2008 dataset.

12 Assuming that you do not want the variable at the end of all the other variables, insert a new numeric variable, equal in size to the newly added variable where you want that newly added variable to be. With small datasets, and in Data View, select the cells with the name of the newly added variable, and select ‘Edit’ from the main SPSS menu, followed by ‘Copy’. Select the cell containing the name of the variable which is to receive the newly added variable and select ‘Edit’ and ‘Paste’ highlight Give it a meaningful name (e.g. pmainsen08). Do not use this Windows copy and paste approach with large datasets. It will simply lock SPSS. Use the Compute procedure explained in earlier Sections.

13 This does not copy the value labels from the new variable. Go to the ‘Variables View’ aspect of SPSS and click on the ‘values’ cell of the newly created variable. Click on ‘Edit’ from the menu at the top of the screen and then on ‘Copy’. (The dialogue box for adding new codes and labels may appear during this process. Close that window). Now click on the values column of pmainsen08 and then click on ‘Edit’ and ‘Paste’ in the ordinary Windows manner.

14 As a check, Crosstabs can be used to crosstabulate the source codes against the new numeric codes and their labels. (Totals should be checked against the frequency table!)

15 If all is well save the file and go on to use or create other lookup tables. If things are not OK……..!

The SEN lookup table is short, and creating a short lookup table should be straightforward. However, lookup tables are not always short. More than 200 separate language codes have been used in the SC since 2007, and creating a lookup table for those codes will take time. Nonetheless, that work is within the bounds of possibility and, once created, the lookup table can be re-used and it may even be possible to adapt it if codes are changed in a later survey. To build scope for change into a lookup table either use numeric codes which increase in greater increments than 1, or give each code several decimal places. New codes, for example 1.5, can be slotted in between existing codes as appropriate and if needed.

The numeric order of the codes is important since it determines the order in which a coded value appears in SPSS output. The order you want may be alphabetical, or reflect some other sort of grouping. In some instances the ordering of items can be contentious. The language lookup file codes languages alphabetically and by region. The regional grouping used in the language lookup file largely but not entirely, follows the UN classification, and some regions have been ‘merged’ where limited pupil numbers suggest that this is appropriate. There should be a reasonable basis for grouping data, and groupings were discussed in advance with specialists working for local authorities. A research analyst’s personal or ideological preferences will not be enough to justify a particular classification. Again, and depending on the project in hand, consultation with research users can be a useful, prudent, step.

Once a file merger has taken place, the file should be saved and any missing values reviewed. The language lookup table includes a string ‘Note’ field to take a short comment on each language. This is unusual in practice at City Hall since lookup tables are used to remove string variables rather than add to them. Nonetheless, the variable is there, and the standard steps for dealing with missing numeric data will not work for the string ‘Note’ field. Two short steps will get round this.

- Compute languagetemp = 1 if note= “ “
- Use Recode into a Different Variable facility so that the target ‘Note’ variable = ‘No note’ if languagetemp=1

This dogleg approach of creating a temporary numeric variable and then using the Recode facility is not rocket science, but it does provide a comparatively quick way of avoiding blank records if string variables are being used.

13. Using Autorecode in creating large lookup tables. Creating a new case, case value and value label

In some instances a variable's existing values can be labelled to add meaning to output, and the free school meal entitlement variable in the NPD was shown as a case in point. In other instances, string variables will have a limited number of codes and the 'Recode into a different variable procedure' can be used to create a numeric equivalent which can, again, be labelled fairly quickly. In the case of the 200 plus string language codes, that procedure would be overly time-consuming, and data have been merged from a pre-prepared SPSS lookup table containing the string codes, withy their coded and labelled numeric equivalents. Creating that particular lookup table was time-consuming, but nonetheless manageable. The table can be re-used with future SC datasets.

In other instances, the number of numeric equivalents needed can be large to the point where none of those procedures would be practicable. Pupil home ward and school name are both cases in point. A ward is a small area of administrative geography, and there are a little under 8,000 wards in England. Few research analysts would wish to create a lookup table for 8,000 wards by typing in a numeric code for each, and then typing in a value label for each code.

While the NPD extract released by DCSF does not contain information on pupil home ward, it is not unique as a dataset in containing information that can be linked to ward information in external files. One such linking variable is pupil home postcode. These have been standardised (See Section 10) and linked to postcodes in other datasets used in Geographic Information Systems (GIS) work in City Hall. These have in turn been linked to 'geographies' from the Office for National Statistics' (ONS), including ward code.

(For those new to ONS geographies, the following links provide useful information and data. http://www.statistics.gov.uk/geography/snac.asp and a beginner's Guide to the geographies used in England is available at http://www.statistics.gov.uk/geography/beginners_Guide.asp)

Mapping data is outside the scope of this Briefing, though it can be a valuable way of presenting information to a reader. However, simply being able to produce a statistical table listing the incidence of A or B in different parts of a region or neighbourhood can also be useful to the reader.
Figure 43 shows an abridged ward file, which, for purposes of illustration, contains only three variables. The first is the name of the ward in text, and the second is the ONS ward code. Figures 41 and 43 also shows a 'count' variable, which in this instance contains the value 1 (one). Figure 43 also shows the coded and labelled variable which will be created through the Autorecode procedure.

Selecting 'Transform' from the SPSS main menu at the top of the screen, followed by 'Autorecode' from the resulting dropdown menu. This leads to the Automatic recode window shown in Figure 41. The left pane of this dialogue box shows the variables in the dataset. Left click on the variable containing the full ward name in text, and then click on the 'arrow' button pointing to the right of the variable list pane. This transfers the name of variable to the 'Variable -> New Name' pane to the right, as shown in Figure 42.

In the 'New Name' of the dialogue box, type in the name of what is to be the new numeric labelled equivalent of the text version of the ward name. Then left click on the 'Add New Name' button below the 'New Name' pane shown in Figure 42. This transfers the new variable name to 'Variable -> New Name' section of the dialogue box. Assuming that you wish the ward names to appear in alphabetical order, select 'Recode Starting from – Lowest value', and then left click on the 'OK' button.

SPSS will create a numeric equivalent of the text version of the ward name, and label that numeric equivalent with the ward's name as shown in the string variable. In this worked example, wards have been arranged in ascending alphabetical order.

Figure 43 shows the 'Data View' window of the SPSS ward lookup table. Abbey Road Ward (and yes, it is *that* Abbey Road) is highlighted, the long windowpane above the list of cases shows that Abbey Road has been given the value 1: it is first on the list alphabetically. The next ward in alphabetical terms (Abbey Ward, all six of them) has been given the new numeric code 2. There are six different instances of that ward name in the dataset, listed against different ward codes. 'Abbey Wood Ward' is a common ward name, and has been given the number 3 and so on. The numbers generated by the autorecode procedure are in the same order as the alphabetical list of ward names, and that sequence determines the order in which the autorecoded version of ward names will appear in an SPSS output. What has been created is, essentially, a large lookup table arranged alphabetically.

Figure 41. Recoding string variables automatically

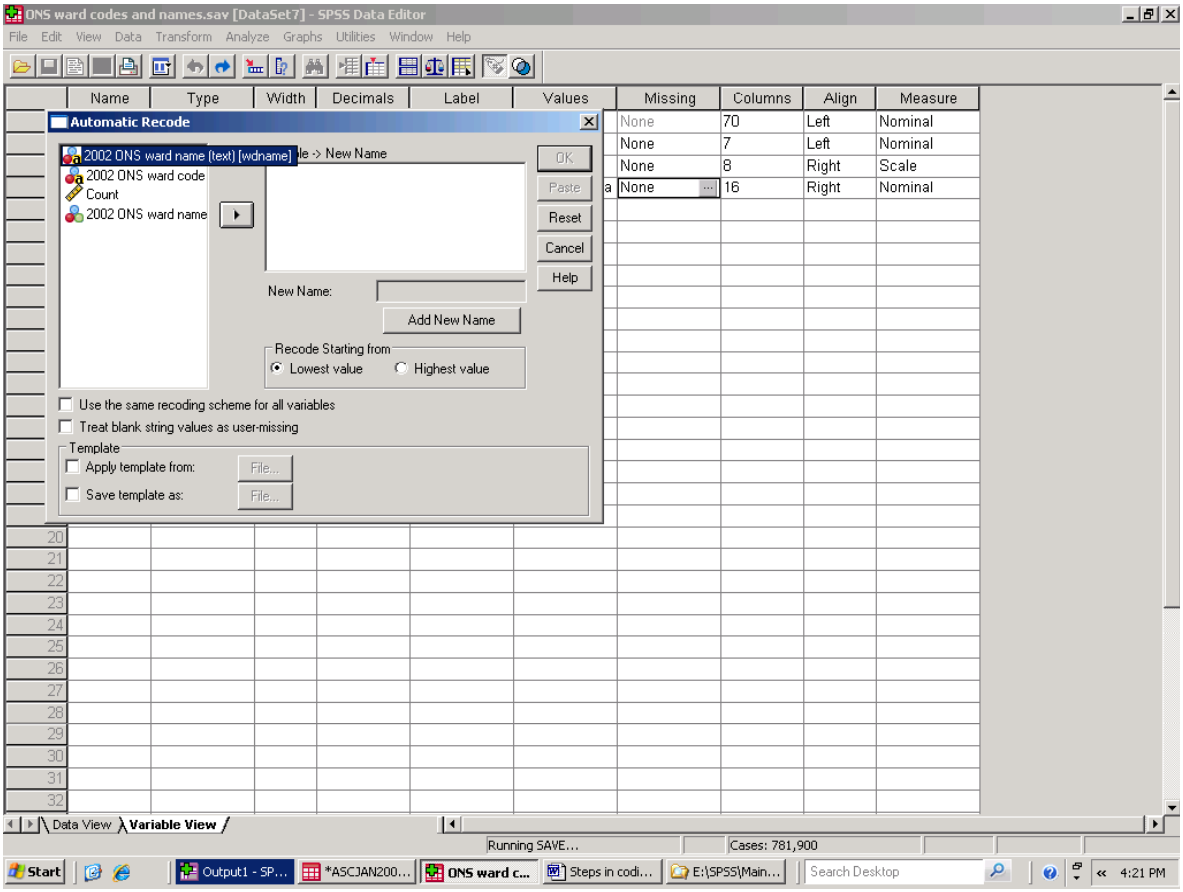


Figure 42. Naming the new ‘autorecode’ variable

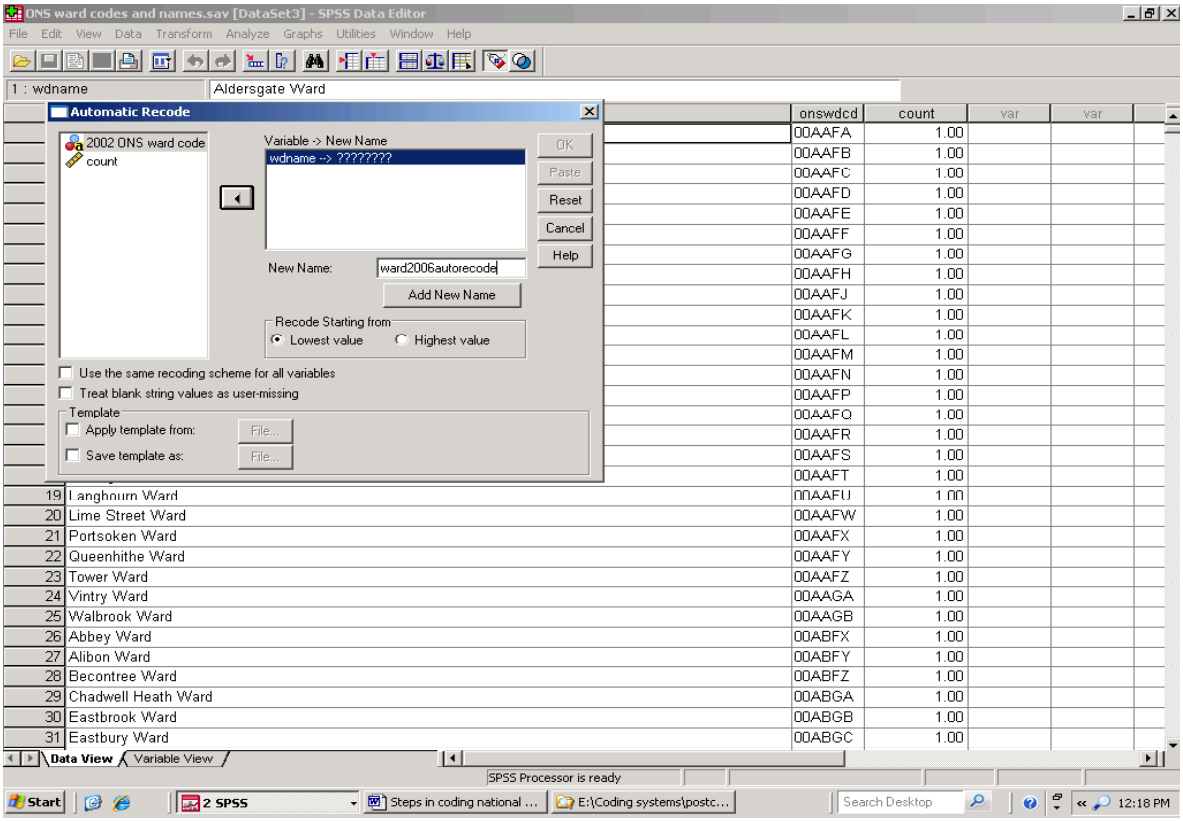
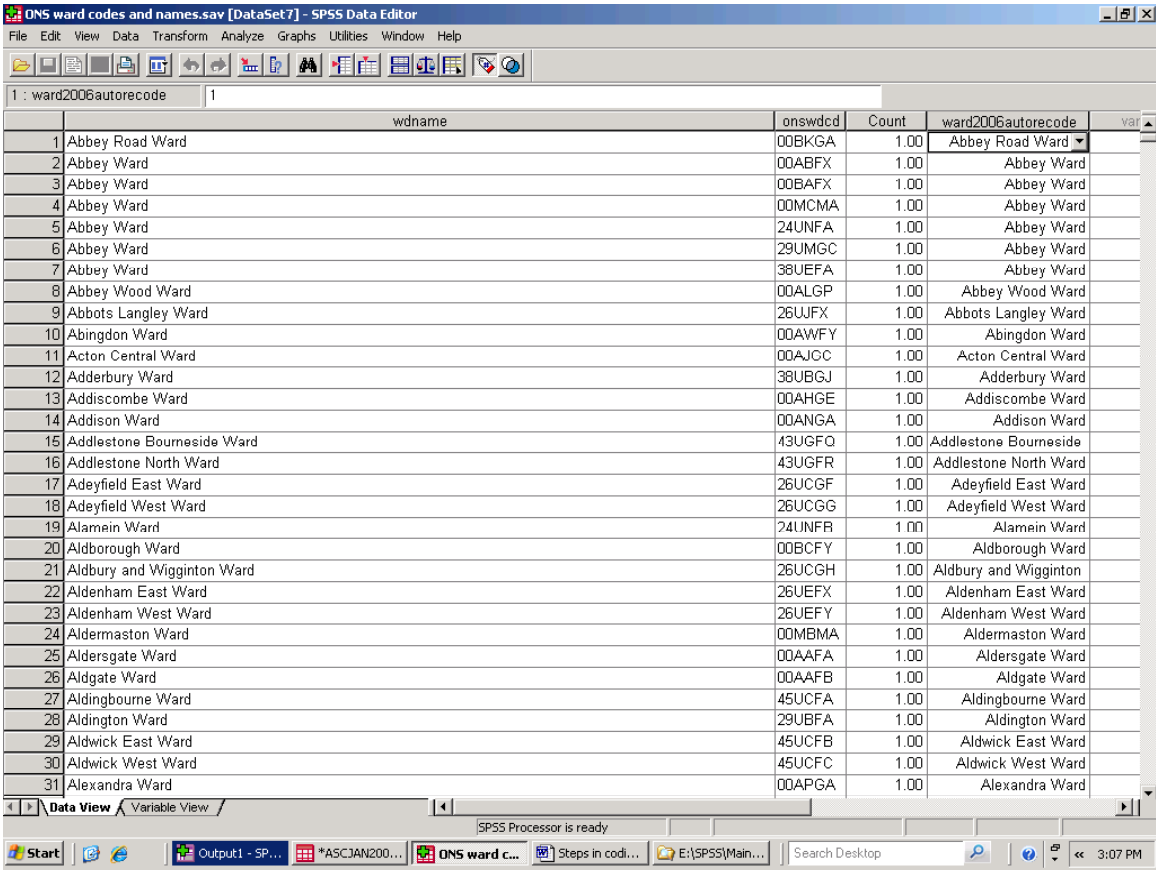


Figure 43. Different wards, same name, same autorecode value and value label



ONS ward codes and names.sav [DataSet7] - SPSS Data Editor

File Edit View Data Transform Analyze Graphs Utilities Window Help

1 : ward2006autorecode | 1

	wdname	onswdcd	Count	ward2006autorecode
1	Abbey Road Ward	00BKGA	1.00	Abbey Road Ward
2	Abbey Ward	00ABFX	1.00	Abbey Ward
3	Abbey Ward	00BAFX	1.00	Abbey Ward
4	Abbey Ward	00MCMA	1.00	Abbey Ward
5	Abbey Ward	24UNFA	1.00	Abbey Ward
6	Abbey Ward	29UMGC	1.00	Abbey Ward
7	Abbey Ward	38UEFA	1.00	Abbey Ward
8	Abbey Wood Ward	00ALGP	1.00	Abbey Wood Ward
9	Abbots Langley Ward	26UJFX	1.00	Abbots Langley Ward
10	Abingdon Ward	00AWFY	1.00	Abingdon Ward
11	Acton Central Ward	00AJCC	1.00	Acton Central Ward
12	Adderbury Ward	38UBGJ	1.00	Adderbury Ward
13	Addiscombe Ward	00AHGE	1.00	Addiscombe Ward
14	Addison Ward	00ANGA	1.00	Addison Ward
15	Addlestone Bourneside Ward	43UGFQ	1.00	Addlestone Bourneside
16	Addlestone North Ward	43UGFR	1.00	Addlestone North Ward
17	Adeyfield East Ward	26UCGF	1.00	Adeyfield East Ward
18	Adeyfield West Ward	26UCGG	1.00	Adeyfield West Ward
19	Alamein Ward	24UNFB	1.00	Alamein Ward
20	Aldborough Ward	00BCFY	1.00	Aldborough Ward
21	Aldbury and Wigginton Ward	26UCGH	1.00	Aldbury and Wigginton
22	Aldenham East Ward	26UEFX	1.00	Aldenham East Ward
23	Aldenham West Ward	26UEFY	1.00	Aldenham West Ward
24	Aldermaston Ward	00MBMA	1.00	Aldermaston Ward
25	Aldersgate Ward	00AAFA	1.00	Aldersgate Ward
26	Aldgate Ward	00AAFB	1.00	Aldgate Ward
27	Aldingbourne Ward	45UCFA	1.00	Aldingbourne Ward
28	Aldington Ward	29UBFA	1.00	Aldington Ward
29	Aldwick East Ward	45UCFB	1.00	Aldwick East Ward
30	Aldwick West Ward	45UCFC	1.00	Aldwick West Ward
31	Alexandra Ward	00APGA	1.00	Alexandra Ward

Data View | Variable View

SPSS Processor is ready

Start | Output1 - SP... | *ASCJAN200... | ONS ward c... | Steps in codi... | E:\SPSS\Main... | Search Desktop | 3:07 PM

So far, so good, but the 'oops' factor can sometimes creep in. Let us assume that somebody (else) has been working with the ward dataset, and has accidentally deleted the record for Alexandra East Ward.

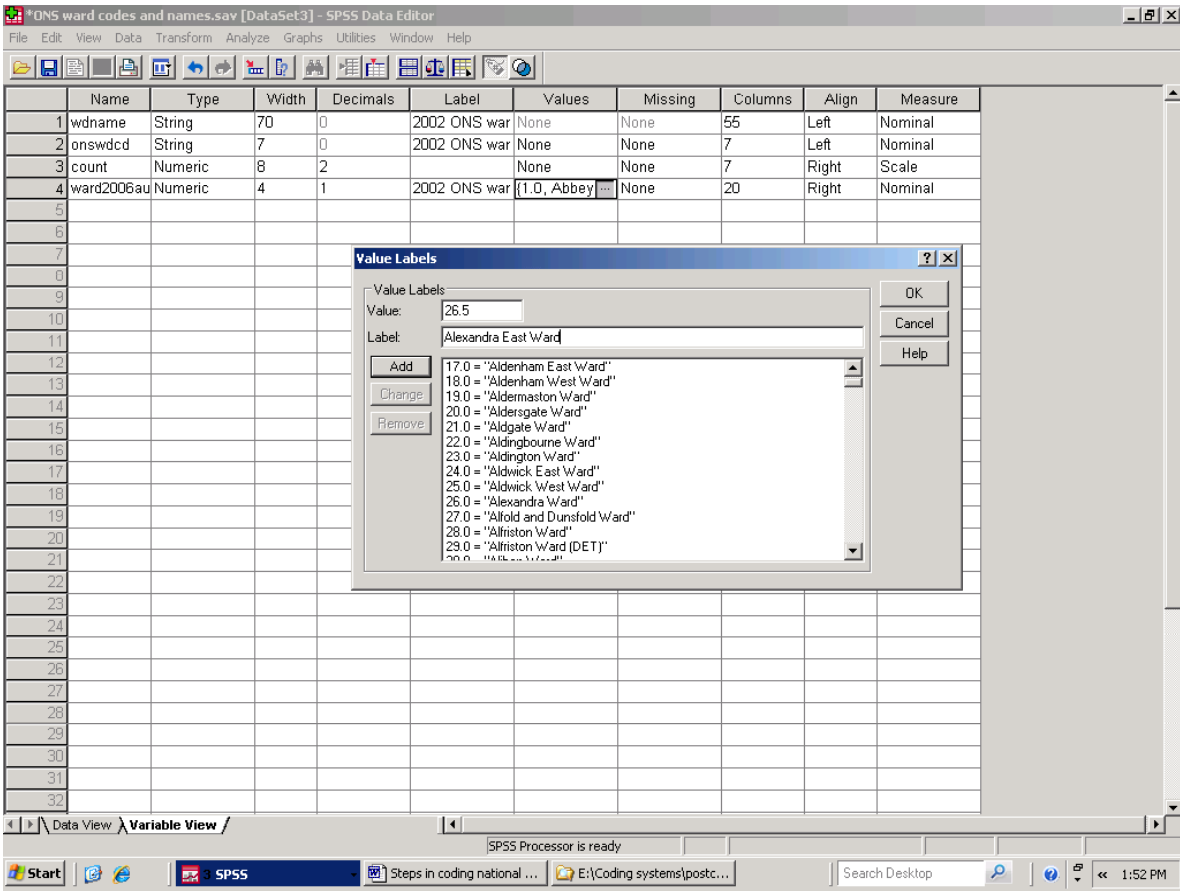
The steps taken to correct this will depend whether the missing variable is discovered before or after a file merge has taken place. Missing variables (and missing values) are best identified and corrected before a file merge takes place, otherwise information gaps have to be made good in both the working file and the lookup table. We will begin here on the assumption that the missing ward has been identified before files have been merged. Much of Section 16 deals with identifying and resolving problematic records in a large lookup dataset.

Alphabetically, Alexandra East Ward should appear between Alexandra Ward and Alford and Dunsford Ward. In lookup table's Data View, select the row containing information for Alford and Dunsford Ward by click on the number to the left of its first variable. Then select 'Edit' in the SPSS main menu, followed by 'Insert Case'. In the new row key in the missing ward name and code.

Alexandra East Ward now needs a new number in the autorecode variable, and an appropriate value label for that number. However, in this instance, autorecode numbers change by an increment of 1, in a numeric variable with no decimal places. In the SPSS Variable View window, click on the 'Decimals' cell of the autorecoded ward variable, and give the variable 1 decimal place. Then left click on the 'Values' cell for that variable, where you will see that Alexandra Ward has a value of 26.0, and Alford and Dunsfold Ward has the value 27.0. Alexandra East Ward fits alphabetically between the two, so key 26.5 in the Value pane of the Value labels window and then key 'Alexandra East Ward' in the label pane. Click the 'Add' button on the left hand side of the Value labels window, and then click the 'OK' button. Return to Data View, and type 26.5 into values cell for Alexandra East Ward. Save the file.

Figure 43 shows six different Abbey Wards in the dataset, each of which have been given the autorecode number 2 and the same value label. You could, in principle, use the procedures just described to edit those numbers so that each ward code in the 'onswdcd' variable shown in Figure 43 has a different number that distinguishes between them. That is not recommended since, as it stands, there is nothing within the ward file to indicate what the order of listing should be i.e. which Alexandra Ward should be listed first, which next and so on. Creating value codes and labels should meet analytical, and not be undertaken slavishly.

Figure 44. Inserting a new value code and value label in a lookup table



We may also want the lookup table to be able to take account of missing values. If, but only if, the variable linking the lookup table to the main dataset it the ward code (in this dataset 'onswdcd'), insert a case with the ward code left blank, and give it an appropriate value label ('Missing data') in the 'ward2006autorecode' variable. Creating a missing value code in this way is appropriate in an alphabetically ordered ward lookup table, where the data are simply nominal (they imply no ranking or measurement). However, as noted on page 24, more care is needed with lookup datasets if data are assumed to be at ordinal level of above.

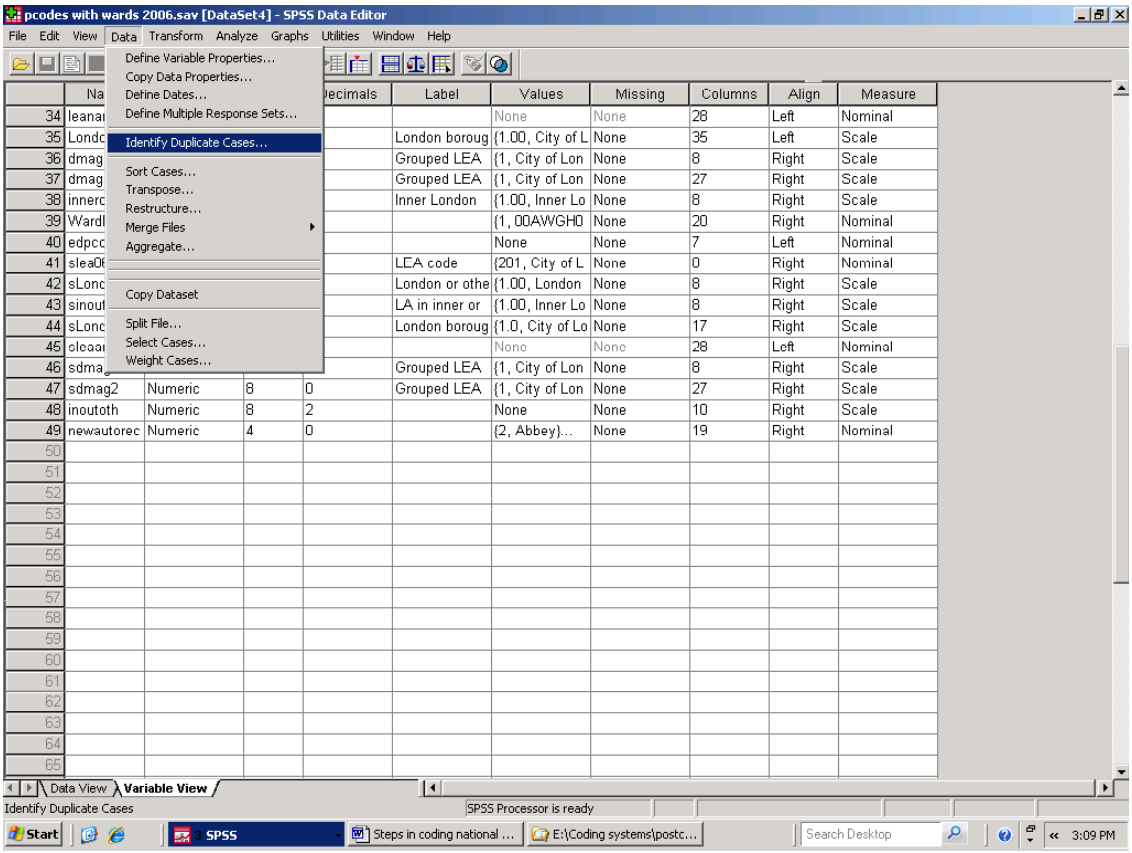
14. Using large lookup tables. Checking for duplicate records and running out of disc space

The edited pupil home postcode is one key variable used to attach information from a postcode and administrative geography dataset to the 2006 pupil dataset. The procedures used are those described in Section 12. An edited postcode variable must exist in the same form in the lookup table and in the pupil dataset (in this instance as a standardised seven character string variable). It must have the same name and width in both datasets, and be sorted in the same way in both datasets. As with previous file mergers, there can be no duplicate postcodes in the lookup table.

With small lookup tables, a visual check should be sufficient to identify any duplicate records. However, there are more than one and three quarter million different postcodes in the postcode dataset, and using visual checks to identify any duplicate records is not a realistic option. Fortunately SPSS has an alternative procedure, which will also work with the larger pupil dataset.

With the appropriate dataset open, select 'Data' from the SPSS main menu, and then select 'Identify Duplicate Cases' on the drop down list.

Figure 45. Identifying duplicate records



In the 'Identify Duplicate Cases' dialogue box that follows, select the variable which will be used to link the lookup table with the main dataset (in this case 'edpcode06'), and transfer this to the 'Define matching cases by:' pane using the right pointing arrow by that pane. Then click the 'OK' button, and a 'RUNNING SORT CASES BY' message will appear at the foot of the screen.

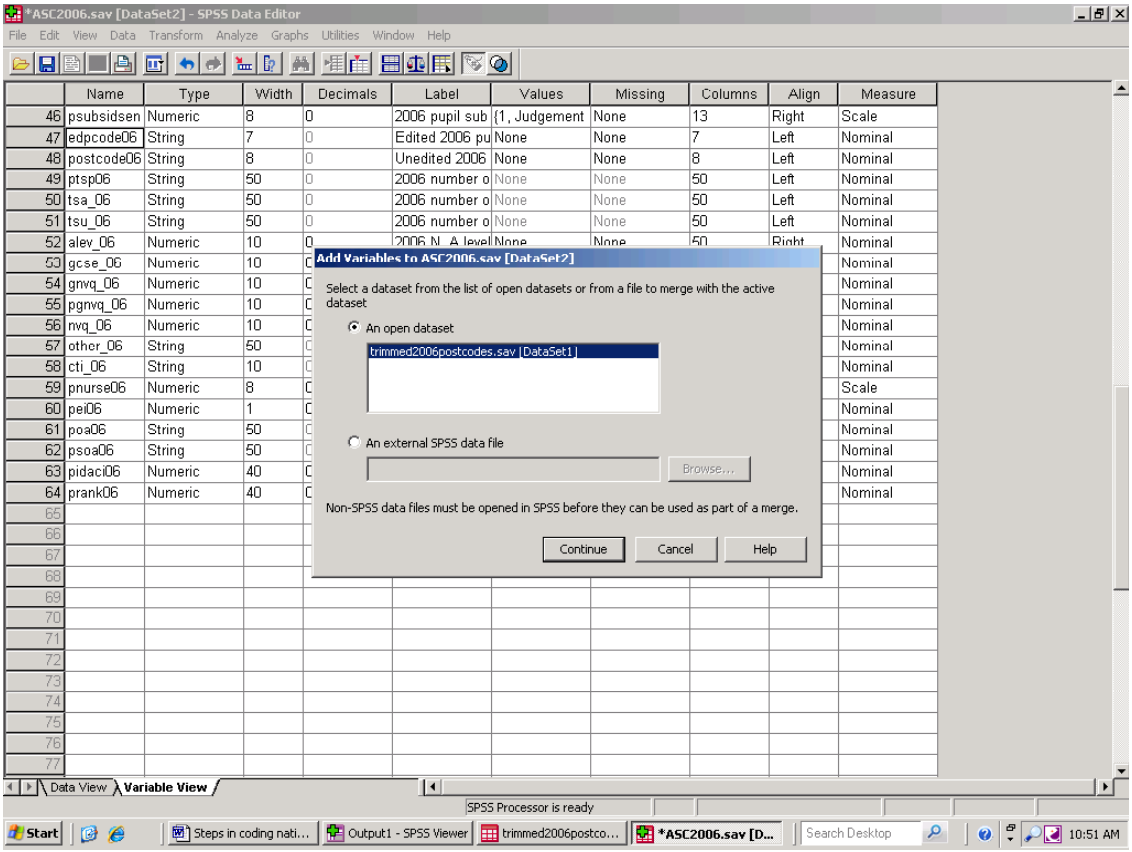
SPSS creates a numeric 'PrimaryLast', in which 0 equals a duplicate case and 1 equals the primary case. Importantly, where a record is identified as a duplicate, this does not mean that SPSS has been able to distinguish records which are 'incorrect' from records which are 'correct'. For example, in the case of there being two N10 9AA records, the first one encountered will be labelled as the primary record, and the second one as a duplicate record. This doesn't particularly matter if the duplicates are duplicates in every respect. Surplus records can simply be discarded using the 'Select if' procedure to retain records with a PrimaryLast value of 1 (see Section 5).

However, there will be a problem if records with the same postcode otherwise have different information, and for example, place the postcode in different boroughs. You may be able to triangulate conflicting records with records in another file, and triangulation is discussed further in Section 16. For the present, we can note that

there are, conveniently, no duplicate postcode records.

Assuming that link postcode variable has already been sorted in the postcode file and the pupil file, we can take a slightly different route from that taken in earlier Sections. Have *both* files open but with the pupil file as the active file. The steps are otherwise as before i.e. Data – Merge Files – Add Variables. This will lead you to the window below where the 'An open dataset' radio button has itself been selected. You can now select the postcode dataset, after which you will select the link variable. Ensure that the external file is flagged as a keyed table (i.e. a lookup table) and begin adding geographic information to the pupil dataset ……. but

Figure 46. Adding variables from an open dataset



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
46	psubsidsen	Numeric	8	0	2006 pupil sub	{1, Judgement	None	13	Right	Scale
47	edpcode06	String	7	0	Edited 2006 pu	None	None	7	Left	Nominal
48	postcode06	String	8	0	Unedited 2006	None	None	8	Left	Nominal
49	ptsp06	String	50	0	2006 number o	None	None	50	Left	Nominal
50	tsa_06	String	50	0	2006 number o	None	None	50	Left	Nominal
51	tsu_06	String	50	0	2006 number o	None	None	50	Left	Nominal
52	alev_06	Numeric	10	0	2006 N_A level	None	None	50	Right	Nominal
53	gcse_06	Numeric	10							Nominal
54	gnvq_06	Numeric	10							Nominal
55	pgnvq_06	Numeric	10							Nominal
56	nvq_06	Numeric	10							Nominal
57	other_06	String	50							Nominal
58	cti_06	String	10							Nominal
59	pnurse06	Numeric	8							Scale
60	pei06	Numeric	1							Nominal
61	poa06	String	50							Nominal
62	psoa06	String	50							Nominal
63	pidaci06	Numeric	40							Nominal
64	prank06	Numeric	40							Nominal
65										
66										
67										
68										
69										
70										
71										
72										
73										
74										
75										
76										
77										

When SPSS merges two files it creates a temporary third file, akin to a dBase 111+ index file. In pc versions of SPSS the file is, by default, written to the C drive. The temporary file will not be visible to you, but it takes up disc space. Assume that the temporary file is at least as large of the two files that are to be merged, and check whether you have that space. If variables from one large dataset are being added to another large dataset, the temporary file can be very large indeed. SPSS may run out of disc space, and the file merge will abort.

If there is a shortage of disc space in computer which is oriented to the C drive, there may be a way out. However, this can be tricky, and it is potentially disastrous. The best route is to consult an IT specialist if work is being carried out on a computer owned by others (and even if it is owned by the user). However in those, possibly rare, cases where a computer only has a C drive, has been used for some time, and has become cluttered with files that should have been deleted in the past, users with access to the drive can delete files to free up space. This is a high-risk approach since files needed to make that type of computer work are generally stored on the C drive. If operating files are deleted, the computer will be out of action until the files are restored, and that can be a very lengthy procedure – assuming that the appropriate backup files exist. As a basic rule, if you have access to the C drive, do not delete anything on it unless know exactly what it is, and are completely sure that it is not needed by the computer or by anyone who uses that computer. If in doubt, seek authoritative advice.

A safer route with that type of computer is to free up more space using 'Disc clean up' and the 'Defragment' facilities. If you have access, you can reach these on a pc by selecting 'All Programs' on the Windows Start menu, followed by 'Accessories', then 'System Tools' and then

'Disc Cleanup' or 'Disc Defragment' as appropriate. If the pc being used is not your own, you may not have access to the C drive, which, frankly, is the safest position for you to be in. IT specialists will have better means of dealing with this issue. Again, seek authoritative advice.

Depending on how the computer being used is set up, you *may* be able to re-direct the temporary file to a different drive with sufficient space to hold it. With SPSS open, click on 'Edit' in the main menu at the top of the screen, and then on 'Options' in the Edit drop down list. The resulting options window is organised rather like a card index or a filing cabinet. The tabs to different sections show different options. Left click on the 'General' tab. This is shown in Figure 48. The 'temporary directory' pane is shown on the left and in the lower part of the tab. In this case, it is set as E:\SPSS\temp – that is, temporary files have been directed away from the C drive.

Personal computers do not usually 'arrive' with separate E drives and if the problem is one of insufficient disc space, you may well need a new hard disc and specialist help to ensure that it and the data and programmes needed are installed properly. An additional useful step at this point may be to have a separate drive created, which holds data and other documents in one place. If the computing system as a whole is changed at some point in the future, data files can be backed up and transferred to a new machine. However, if you are working within an organisation it is clearly important that its ground rules on deleting and copying files are observed, including the ground rules on data confidentiality. If you do not know what those rules are, find out. Remember that, in some contexts, tinkering with drives and directories, and copying confidential files is a very serious offence indeed.

Figure 47. Disc Cleanup and Disc Defragmenter

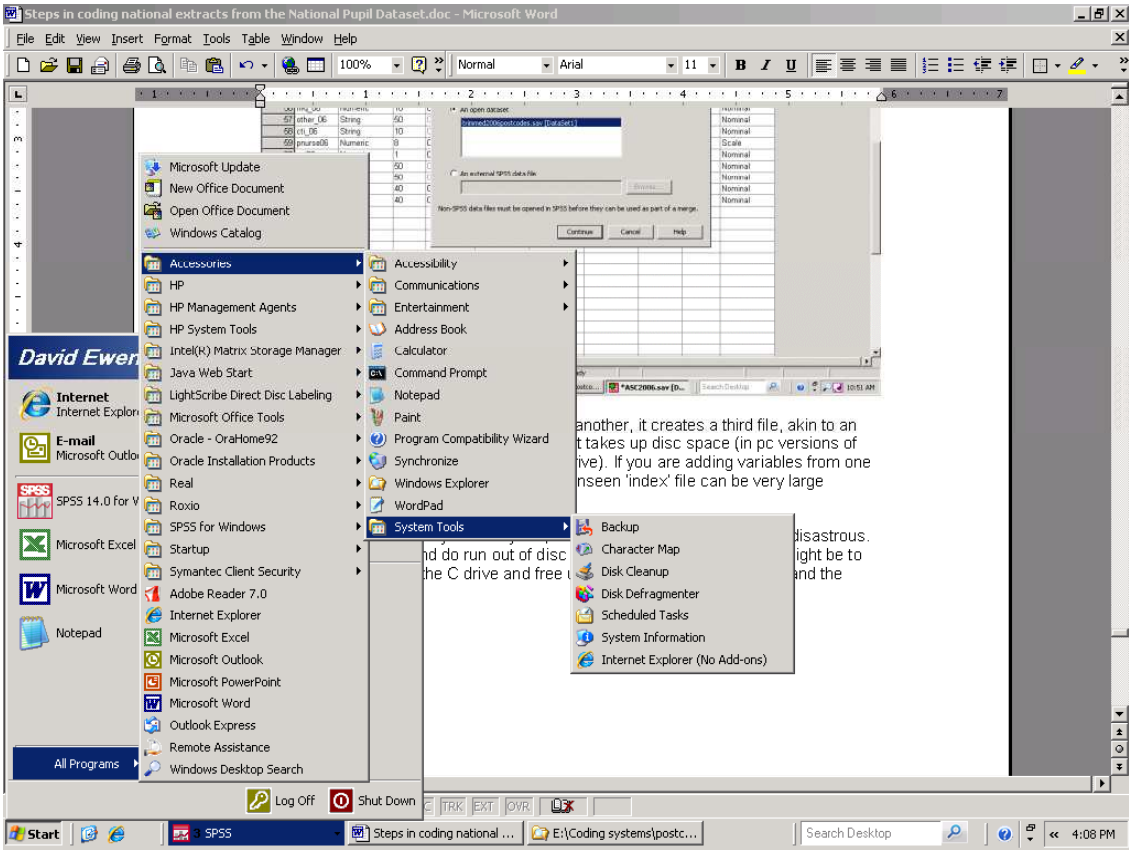
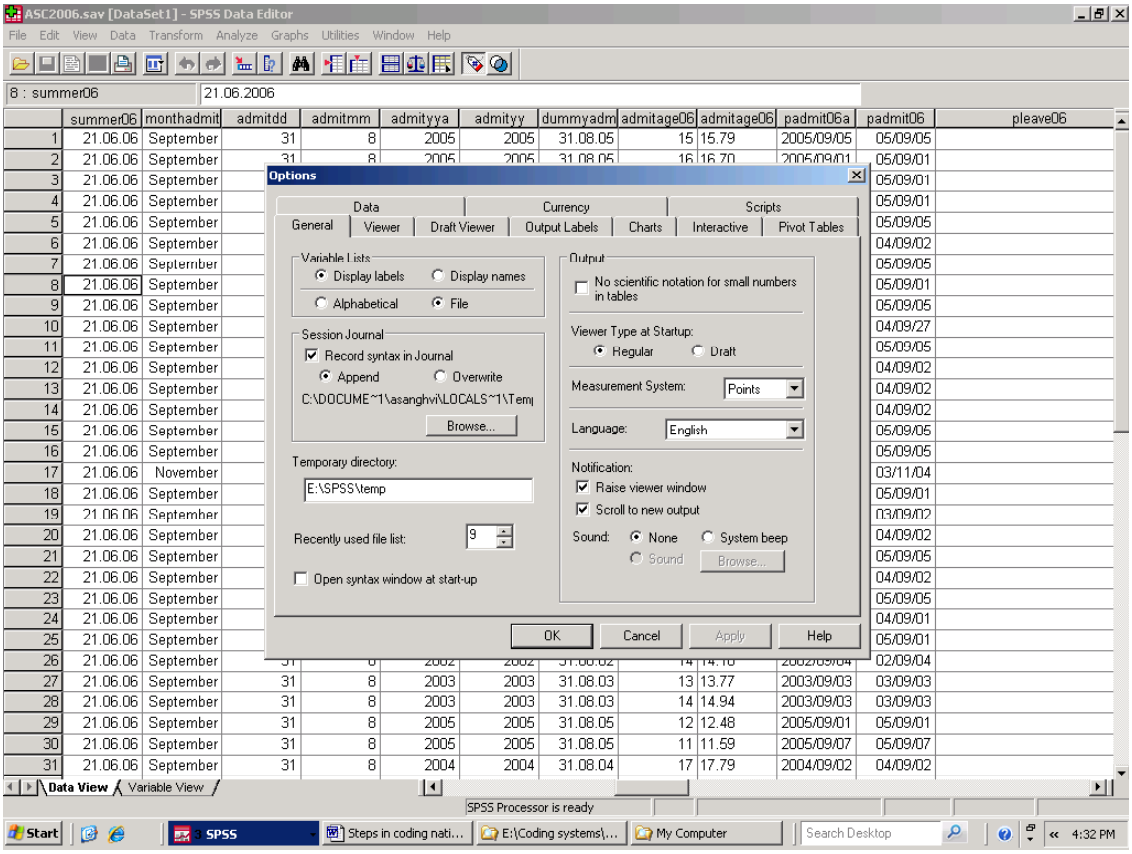


Figure 48. Re-directing temporary SPSS files away from the C drive



15. Merging pupil datasets from different years – missing unique identifiers and a hidden variable

Assume that the user wishes to merge NPD files, containing individuals' records from two School Census. As with previous files mergers, any two files will contain the same 'link' variable. In a school dataset, each school would have a unique identifier, and in a pupil dataset each pupil would have a unique identifier. An alternative involves matching pupil records using several variables to link records of individuals who have the same name, gender and date of birth. This is sometimes referred to as 'fuzzy matching', and the word 'fuzzy' is apt. In a dataset for a single school, let alone in a national dataset, there could be no guarantee that a file merger based on name, gender and date of birth could/would match records for the same individuals. Most database packages will allow for fuzzy merging, and database packages tend to be more flexible than SPSS in this respect. However, SPSS will allow file mergers based on more than one variable, subject to the constraints that have already been described.

Where it can be arranged, file merger based on a single unique identifier is clearly preferable to fuzzy matching. The NPD contains unique pupil numbers (UPNs). These are not ordinarily released to researchers, though DCSF does release pseudo-UPNs. This is held in variable named 'pmr', which can be used to link pupil records from the same year in the files shown in Figure 1. The pmr variable can also be used in the procedures to link pupil records from different years. Ideally, no pupil record would lack a unique identifier, since SPSS will read two or more blank pmr records as duplicate records and that would prevent a file merge.

Extracts from the NPD have been received at City Hall for each year from 2002 to 2008. From 2002 to 2004, all pupil records had a unique pmr. From 2005, the extracts were ultimately national in coverage. In 2005, some pupil pmr records were blank. For the main part, this reflected pupil turnover in part-time nursery provision outside London. This results in uncertainly about whom existing UPNs in the nursery class record actually refer to. To avoid attributing a unique ID to the wrong pupil, DCSF left those pmr records blank.

Fortunately, it is comparatively easy to create new unique codes to fill that 2005 gap. However, the pmr variable is the single key linking variable for attaching data to the SC file from the assessment files for the same year, and from SC files for other years. Given its critical importance, work is carried out on a copy of the pmr variable, on a 'pseudo pmr' to avoid the risk of the source data being corrupted. If in doubt, take a backup copy of the dataset at this point.

Sorting the 2005 pupil dataset in ascending order on the pseudo pmr variable places blank pmr pupil records at the top of the dataset. Other pseudo pmr records are placed in alphabetical order. With that done, a new numeric variable (ptemp05) can be given the value 1 whenever the pseudo pmr is blank. This is achieved by

Compute ptemp05=1 if pmr=" "

A frequency table run on the variable 'ptemp05' shows how many pupil records had blank pmr records, and this figure should be recorded in writing in a daybook. We will refer to that figure as 'A'. (It goes without saying that the file is being saved at intervals).

At this point, a numeric spss ID (pspssid05) variable is created large enough to give a unique numeric code to each pupil record in the 7.5 million-pupil dataset. SPSS datasets contain a hidden '$CASENUM' variable which automatically gives a unique number to each case/record, with the first case listed being given the number 1. That value changes if the position of a record in the dataset changes.

In the next two steps

Compute pspssid05 = $CASENUM

and then convert spssid to a string variable.

The next step is

Compute copy of pmr variable=pspssid05 if temp=1

Finally, convert pspssid05 back to a numeric variable.

The numeric pspssid05 variable follows exactly the same order as pseudo pmr variable, and sorting on the former has the same effect as sorting on the latter. This is useful, since SPSS prefers to work, and tends to work faster with, numbers, Thereafter, whenever the dataset needed to be sorted on the pseudo pmr variable, for example to bring in assessment data, it can now be sorted on the pspssid05 variable. However, is there a shorter way of doing things?

There is a more elegant way of creating a string version of the spssid variable using $CASENUM. It does not save a great deal of time, and it is mentioned here mainly because it points to ways

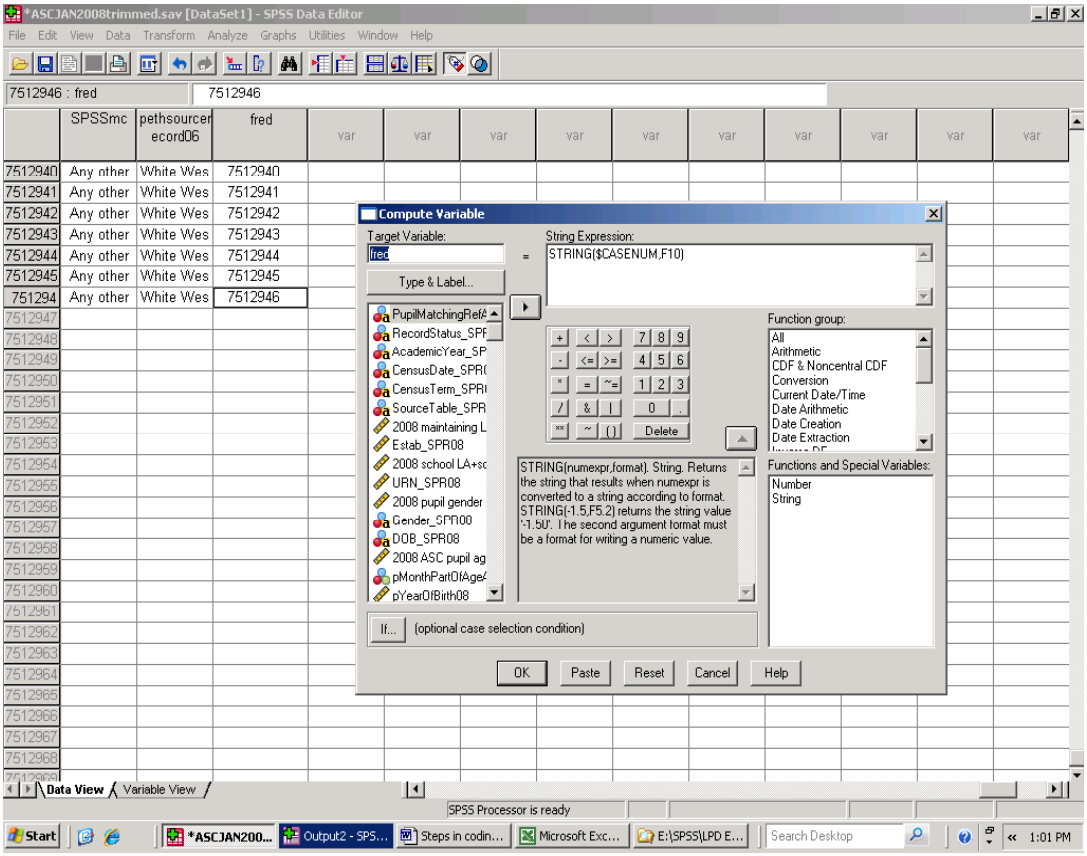
in which the user can develop skills in SPSS independently.

The SPSS Compute Variable window, shown in Figure 49, includes a Function Group facility, with options listed on the right of the dialogue box. The functions 'Number' and 'String' are options in the 'Conversion' function group. There is a brief explanation to the left of the 'Functions and Special Variables' section of the dialogue box, and the upward pointing arrow button above that explanation can be used to put the prefix 'STRING' into the String Expression Window.

The default format for a numeric value is F8.2, which in English means that the variables are eight characters wide, *including* two decimal places. Figure 49 shows a shortcut to creating a string version of the $CASENUM variable using that string function, with the string variable set to F10, that is to 10 characters wide. Figure 49 also show that the facility can also accommodate negative numbers.

There are some 19 function groups, and a far larger number of options. The majority are statistical functions but some, including those relating to dates, are useful in organising data. They are there to be explored (preferably using dummy data to do so, and when and if the day job allows). However, the notes can work better as a reminder for those who already know the procedures involved, than they do as a guide for the beginner. Where you learn more about a new SPSS procedure, make notes in a daybook for future reference.

Figure 49. Creating a string version of a numeric unique identifier



The steps above are not difficult, and will remove multiple blank (and therefore duplicate) pmrs *in the situation described.* However, there will be a major problem if exactly the same procedure is used to replace missing data in the 2006 pseudo pmr record with a view to merging the 2005 and 2006 datasets using the pseudo pmr variable as the link variable.

If the dollar $CAENUM facility is used as described earlier in the Section, the first pupil record with a blank pmr in the 2005 pupil file, and the first pupil record with a blank pmr record in the 2006 file would both be given the pseudo pmr '1'.

However, precisely because those two records originally had blank pmrs, there is no way of knowing whether the records actually are for the same individual. To avoid merging records of potentially different people, ensure that none of new numbers used to replace blank pmr records in one-year duplicate numbers used to replace blank pmr records in the previous year.

The opening steps of work with the 2006 pupil dataset the opening steps were as described earlier. The pmr variable was copied to a pseudo pmr variable, and the dataset was sorted on that pseudo pmr variable. A 'temp06' variable was

created and given the value 1 when the pseudo pmr record was blank. However, the next step (Compute spssid-$CASENUM) was modified.

The total number of pupils who originally had blank pseudo PMR records in the 2005 data was 'A' (see page 58). In ordinary circumstances the first value for $CASENUM is 1, the second is 2 and so on. Adding 'A' and 1 gives a number which is one (1) larger, and therefore different from, the largest $CASENUM value computed for the 2005 pupil file. The next step therefore is

Compute spssid06 = $CASENUM+A

Running a frequency Table on temp06, provides the total number of pupils records with blank pseudo pmr values. We will call this 'B'.

On this basis, spssid07 will be arrived at as

Compute spssid07= $CASENUM+ A+B

That procedure can continue with 2008 and 2009 data, and beyond, to ensure that there are no false matches between pupil records in one year and another. However, if the prospect of creating what would ultimately be a very large spssid value appears daunting, consider the prospect of creating a full longitudinal dataset, with more then 7 million pupil records each year!

Figure 50 gives a record of FSM entitlement from 2002 to 2005, and gives a more realistic picture of the situation over time than we would have by comparing the record for 2002 and 2005 alone. Longitudinal data are of considerable value. We live our lives in time, rather than as a series of snapshots revealed by cross-sectional surveys. While a discussion of the value of longitudinal studies is beyond the scope of the Guide, there are a number of websites that provide further information. These include

- Longview at http://www.longviewuk.com/
- The Centre for Longitudinal Studies at http://www.cls.ioe.ac.uk/
- The UK Longitudinal Studies Centre at http://www.iser.essex.ac.uk/survey/ulsc
- and UK Data Archives at http://www.data-archive.ac.uk/
- ESDS, and joint service from UK Data Archives and the UK Longitudinal Studies Centre, at http://www.esds.ac.uk/longitudinal/about/introduction.asp
- The Economic and Social Research Council (ESRC) Census Programme at http://census.ac.uk/guides/Longitudinal.aspx#3

Issues around records which are not blank, but which are nonetheless duplicates remain. The 'identify duplicate records' sorts data, and groups duplicate records next to the pupil record each duplicates. It *may* be possible resolve difficulties by scanning records visually in SPSS Data View. For example, if the concern is with pupils on roll in any one January, and a particular pupil has a record from a school which he or she left in the previous summer, who also has a current (January) record with the same pmr, then the record of the summer leaver can be deleted. While that is a real world example drawn from work at City Hall, resolving duplicate records may not always be so easy, and the next Section outlines the way in which the triangulation of data can help the user choose which record to work with, and improve data quality more generally in 'live' datasets

.

Figure 50. The longitudinal record of free school meal eligibility

	Years (January) for which pupils were recorded as entitled to free school meals																
	2002, 2003, 2004 and 2005	2003, 2004 and 2005 only	2004 and 2005 only	2002, 2003 and 2004 only	2003 and 2004 only	2002 and 2003 only	2005 only	2004 only	2003 only	2002 only	2002, 2003 and 2005 only	2002, 2004 and 2005 only	2003 and 2005 only	2002 and 2004 only	2002 and 2005 only	Not entitled to FSM in any January 2002-2005	Total
Roll status 2002 2005																	
Number																	
On roll 2002, 2003, 2004 and 2005	126,403	24,588	14,800	18,685	5,050	14,238	12,302	4,882	5,195	17,342	6,107	7,016	1,809	1,810	2,847	496,062	759,136
On roll 2003, 2004 and 2005 only		14,254	11,344		1,994		4,930	2,037	1,724				865			68,331	105,479
On roll 2004 and 2005 only			17,719				14,512	3,106								79,421	114,758
On roll in 2005 only							113,307									974,877	1,088,184
On roll 2002 2003 and 2004 only				15,935	2,155	3,371		1,469	825	3,198				869		64,403	92,225
On roll 2003 and 2004 only					1,307			613	301							6,254	8,475
On roll 2002 and 2003 only						20,824			3,674	5,368						79,072	108,938
On roll in 2004 only								2,323								10,998	13,321
On roll in 2003 only									2,844							12,672	15,516
On roll in 2002 only										27,303						99,378	126,681
Total	126,403	38,842	43,863	34,620	10,506	38,433	145,051	14,430	14,563	53,211	6,107	7,016	2,674	2,679	2,847	1,891,468	2,432,713
Source: merged 2002, 2003, 2004 and 2005 LPD																	
Percentage																	
On roll 2002, 2003, 2004 and 2005	16.7	3.2	1.9	2.5	0.7	1.9	1.6	0.6	0.7	2.3	0.8	0.9	0.2	0.2	0.4	65.3	100.0
On roll 2003, 2004 and 2005 only		13.5	10.8		1.9		4.7	1.9	1.6				0.8			64.8	100.0
On roll 2004 and 2005 only			15.4				12.6	2.7								69.2	100.0
On roll in 2005 only							10.4									89.6	100.0
On roll 2002 2003 and 2004 only				17.3	2.3	3.7		1.6	0.9	3.5				0.9		69.8	100.0
On roll 2003 and 2004 only					15.4			7.2	3.6							73.8	100.0
On roll 2002 and 2003 only						19.1			3.4	4.9						72.6	100.0
On roll in 2004 only								17.4								82.6	100.0
On roll in 2003 only									18.3							81.7	100.0
On roll in 2002 only										21.6						78.4	100.0
Total	5.2	1.6	1.8	1.4	0.4	1.6	6.0	0.6	0.6	2.2	0.3	0.3	0.1	0.1	0.1	77.8	100.0

Source: DMAG Education, merged 2002, 2003, 2004 and 2005 LPD

16. Using a 'live' dataset as a lookup table. Risks and triangulating with other datasets to reduce the risk of error

EduBase is a national education institution dataset. It is the single best source of national information on a wide range of institutional variables of interest in education research and statistics. A number of those variables are added to pupil level datasets in work at City Hall. EduBase is a 'live' dataset, in the sense that updating is ongoing, using information provided in part by schools, rather than being a one-off snapshot. This Section uses work with EduBase to bring together a number of points made in earlier Sections, and to introduce situations that can arise in work with 'live' datasets, and which will need to be resolved.

EduBase data extracts are accessed as a csv (comma separated values) file. SPSS can read a csv file directly, and it can read an EXCEL version of a csv file. Section 4 sets out how to open both types of file in SPSS. Whether it is practicable to transfer a csv file to EXCEL and then to SPSS depends on the number of cases (rows) involved. EXCEL 2000 does not allow enough rows to take a full EduBase file and this may apply to other datasets users may wish to open in EXCEL.

As with the postcode and pupil datasets, many of the string variables in EduBase need to be recoded into numeric equivalents, with value labels added. Additionally, information has been added to the EduBase extract used in City Hall so that London local authority names appear in alphabetical order, with inner London boroughs listed first. It is quicker to add that information to the EduBase extract before it is merged with the pupil dataset, simply because the former is considerably smaller than the latter, and takes less time to process.

Adding variables from a lookup table to a main dataset will increase both file size and the time required for data processing. If there are variables that are not needed in the lookup table, delete them (working with a copy of the file). In the SPSS Variable View window of the SPSS dataset, simply select the variables to be deleted by left clicking the number of the variable on the left hand side of the window and, once more, press the delete button on the computer keyboard. The same considerations and choices are likely to apply to other datasets. If information is to be added from one file to another, do not add unneeded data.

The procedures for merging EduBase and the pupil dataset are the same as those described earlier Sections. The merger requires a link variable, common to the datasets in question, and there is no such variable in EduBase at the outset. However, a link variable can be created from *two* existing variables, which exist in EduBase and in the pupil datasets.

Each education institution has a three digit LEA number and a four digit institution number. These can be combined to make what those who have worked on school data may expect will be a unique institution school identifier. That combined number provides the link with the pupil dataset. To create the link variable

Compute schlidyy = (LEA number*10000) + institution number.

EduBase is an education institution dataset, rather than a maintained school dataset. Educational institutions, which are not maintained schools, may be given the four-digit institution number of 0000 in EduBase. Where there is, for example, more than one university in a local authority area, which is the case in London, these will have the same combined LEA and school number. They will have duplicate 'unique' ids.

In the procedures outlined in the Guide so far, information cannot be added to a dataset from an external lookup file if this has duplicate records in the key, linking, variable. If there are only a few duplicate records, which are in any event not relevant to your analysis, you may be able to decide which records to delete on the basis of a visual scan. To delete a record in the dataset's Data View window, left click on the number of the case shown on the left hand side of the computer screen, and press the delete key on the computer's keyboard. In other instances there may be too many duplicate records for this to be a particularly convenient approach. The solution for research on maintained schools at City Hall is simply to delete records of institutions with a 0000 institution code (which are in any event not maintained schools).

Section 6 sets out how users can delete large numbers of cases. In this instance use Data\Select Cases\ and

Select Cases if the four digit school code is >=1.

in combination with

the 'Delete unselected cases' option in the 'Select Cases' dialogue box (see Figure 10).

Remember to check whether 'Delete unselected cases' is in place before the Select Cases procedure is used again. If it is, but is not needed, switch the radio button off.

These procedures have, as noted, been covered in earlier Sections, and are brought here together in combination for purposes of illustration. Work with EduBase also highlights a new issue.

The SEN and language lookup tables are comparatively small, and were deliberately constructed for use as lookup tables. They are comparatively easy to check, and can be expected to be complete. The postcode and administrative geography dataset is larger, but as long as the person using it has a reasonable grasp of administrative geography, it can be checked using frequency tables and using the procedures set out in Sections 17 and 18. Additionally, the dataset was created under conditions which meant that it was as complete as could reasonably be expected at a particular point in time.

However, EduBase is a 'live dataset', in the sense that it is updated at, perhaps unpredictable, intervals throughout the course of the year, and quite possibly by different individuals in the same school. At any one moment the data may not be fully up to date, complete or accurate. Data entry arrangements are open ended, which leaves considerable scope for human error. Missing data and miscodes are more common than might be expected.

EduBase also contains a wide range of mainly string variables, and some of these may well deal with matters which are outside the individual research analyst's area of expertise, and which he or she will not (initially) find easy to check. In short, the dataset contains pitfalls for the unwary. Those working with EduBase will need to understand the variables involved, and be able to correct for at least some miscodes and missing data. The user may also encounter circumstances where missing data, or where filling information gaps through best guesses based on statistical inference, will not be acceptable.

The scope for correcting information gaps can depend in part on the researcher's access to other datasets, which can be used to triangulate (i.e. check) data from live datasets.

By way of illustration, one of the EduBase variables added to the 2006 pupil dataset was whether the school was a boys' school, a girls' school or whether it had a mixed intake. A frequency table run on the 2006 pupil dataset after the merger showed that 23 pupils, attending the same school, lacked that information. A further check indicated that no information of any sort had been added from the EduBase file to the records of those 23 pupils on roll in 2006.

The pupils had the same unique school code in the original pupil dataset, but this did not match any school code in the EduBase file. The pupils were all in the infant school age range, and had pronounced levels of special educational needs.

Figure 51. School attended in 2006, gender of intake

School intake	Frequency	Percent
Boys	169,590	2.2113
Girls	227,171	2.9622
Mixed	7,272,331	94.8262
Total	7,669,092	99.9997
System missing	23	0.0003
	7,669,115	100.0000

Source: 2006 English Pupil Dataset

The local authority three digit code meant that the local authority could be identified, and a check on the Authority's website confirmed that it maintained a special school for children of infant school age. A school with the same name, but with a different four digit institution code was listed in the 2006 EduBase file. A further check, this time on an EduBase file from 2005, revealed a school with the same and name and local authority as that in the 2006 EduBase file, but with the same institution code as that in the 2006 pupil dataset. That is, the two versions of EduBase 'disagreed' with each other. The school's record had changed over time. This is what might be expected in a 'live' dataset.

This is a simple (but actual) example of triangulating data in one dataset with information from elsewhere. In this instance, with so few pupil records to correct, sorting the data in ascending order on one of the school variables with missing information put the pupil records with missing the data in the variable at the top of the dataset in the SPSS Data View window. Where information was coded, the 2005 SPSS version of EduBase provided *a* version of most codes. These were checked against the codes used in the 2006 dataset and the appropriate code was entered in the top row of blank cells. Those codes were identified from the Values cell for the relevant variable in the Variable View window, and then keyed into the relevant cell in the Data View window (see Figure 52). Alternatively, left clicking on a blank cell in the Data View window will produce a drop down list of the value labels used for that variable. Entering the missing data for one pupil, and then using a spreadsheet-type copy and paste approach, completed the record in an acceptable amount of time.

Filling in missing data will not always be this simple; see Figure 53.

Figure 52. Copy and paste in SPSS Data View

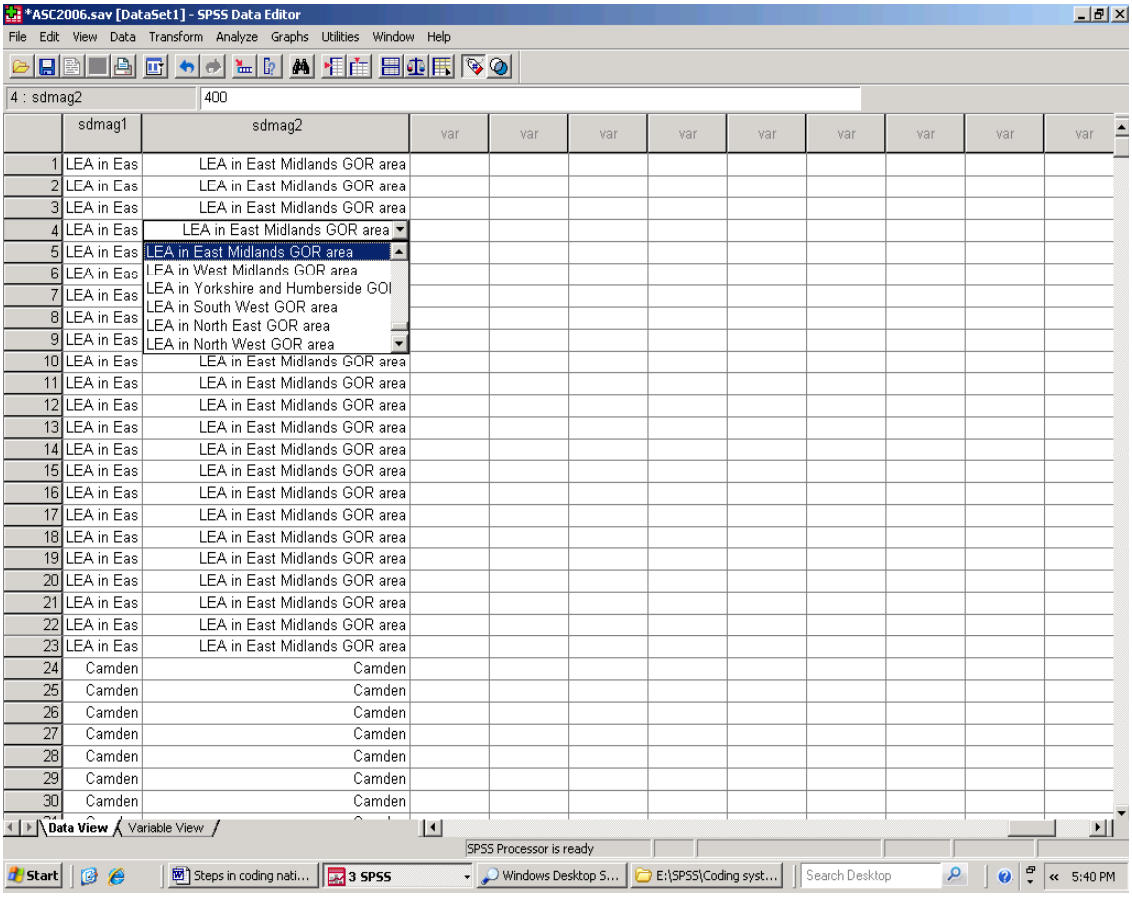
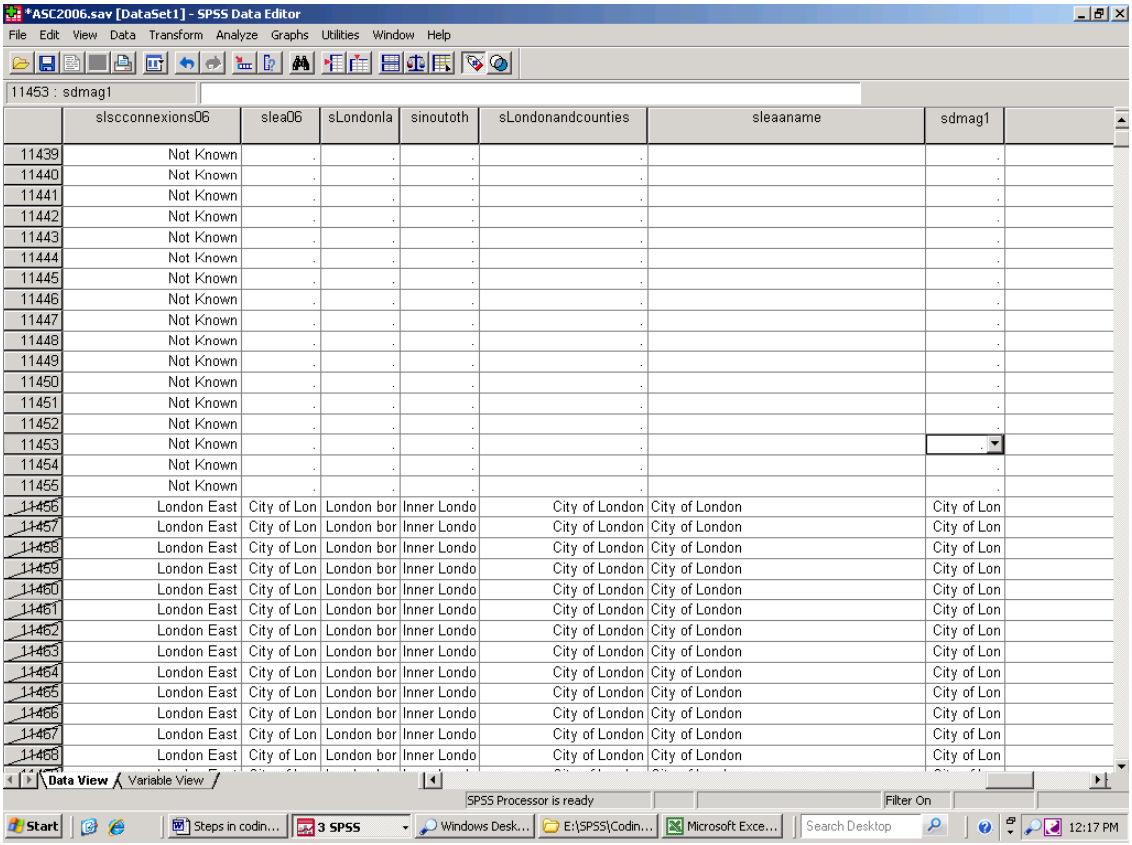


Figure 53. Eleven thousand missing geographies



In 2009, one project using pupil level data required, for the first time, the identification of the local authority district in which each school in England was located. In terms of administrative geography, a district is any one of (a) a part of a county (b) a unitary authority or (c) a London borough. The project was such that missing data in the district variable was unacceptable; the record needed to be complete for each January from 2005 to 2008.

Previous checks on earlier pupil datasets showed, that in 2006, 11,455 pupils had a wide range of missing data relating to the geographical location of the school, including its district and ward. (See Figure 53). In those cases, the autorecoded version of the school ward had all been given the value '1' by SPSS, and '1' does not feature in the list of codes with value labels. It was added by SPSS off its own bat, and signifies 'missing data'.

Selecting records where the school's ward had the value one, and running a series of frequency tables on other school characteristic variables, showed that each school with a missing geography was a British Forces school. These were all reclassified as 'Overseas Schools', with that flag included in the school ward, school district, and school region variables.

The large number of pupils with missing information on the administrative geography of the school attended meant that the copy and paste procedure described earlier in this Section was not appropriate. The gap was filled using SPSS 'Compute' and 'Recode' facilities. The quickest way of resolving that issue would have been to carry out the Compute and Recode exercises on the EduBase extract before it was merged with the pupil file, rather than afterwards.

Current versions of EduBase have been triangulated with

- earlier versions of the same dataset,
- earlier versions of the English Pupil Dataset
- Grid reference information has been triangulated with information from commercial postcode datasets held under license by the GLA,
- administrative area information has been triangulated in National Statistics Standard Names and Codes (SNAC) files.

This was time-consuming, but unavoidable when missing data are simply not acceptable. Again as a general principle, if data are missing in one dataset, replace it with data from a reliable data set where that is available. Where data in different datasets 'disagree', try triangulating with a further trusted source of the relevant information. Regrettably, this will not always provide the answers needed, and checks will need to be made with individuals who might hold that information. Tact is nearly always an asset in those circumstances.

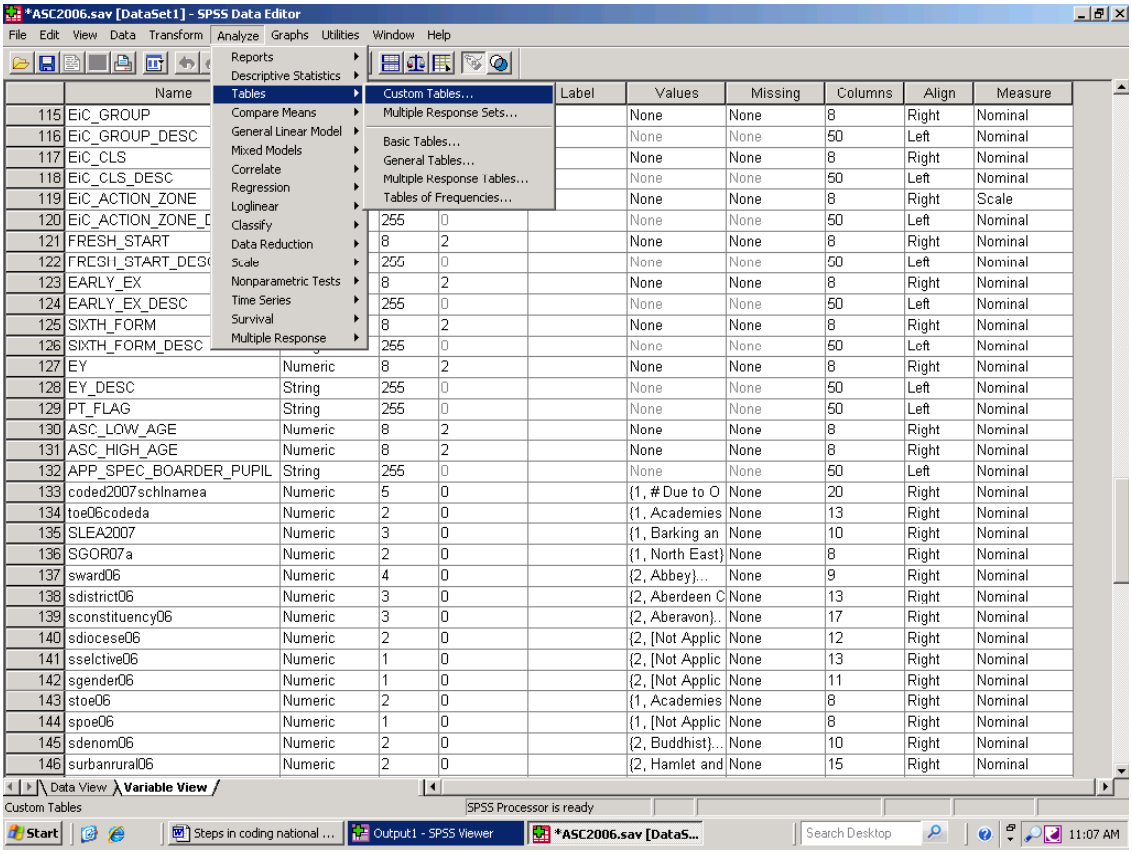
17. Using the Tables facility to check data. Too many values, long string variables, and overloading Tables

Assuming that files have been merged, checked, cleaned up as necessary and saved, the SPSS Tables facility can be used to provide a further quality check on data. The Tables facility can organise data more flexibly than the Crosstabs procedure allows, which in a sense means that data can be subject to tougher scrutiny. It may also be the case that the Tables module can be used to organise data a form needed for analysis by others and/or in other software.

To illustrate this, pupil level data will be grouped in terms of 'home' local authority area, ward and the maintaining local authority for the school attended. The Tables menu is accessed by selecting 'Analysis' on the main menu, and then by selecting 'Tables' from the drop down list.

There are, however, two potential obstacles to this. For purposes of illustration, we will select Custom Tables.

Figure 54. Analyze/Tables

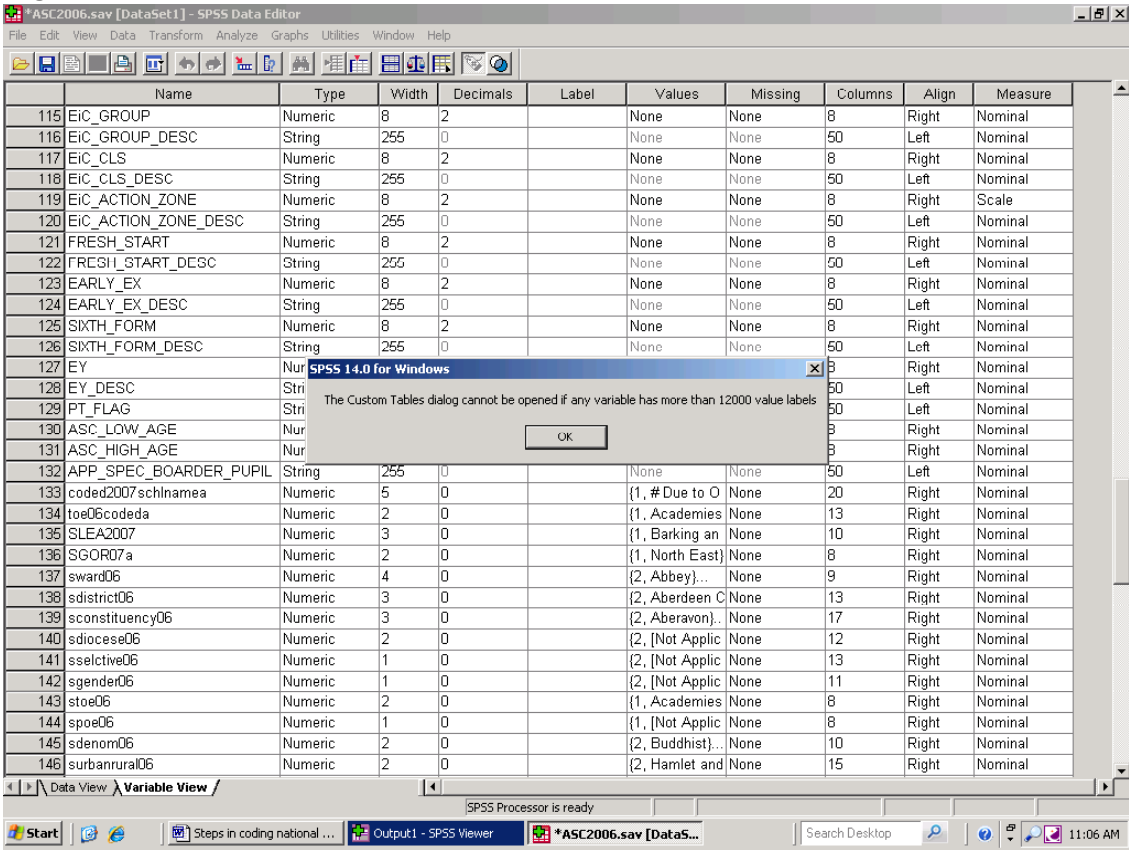


	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
115	EiC_GROUP					None	None	8	Right	Nominal
116	EiC_GROUP_DESC					None	None	50	Left	Nominal
117	EiC_CLS					None	None	8	Right	Nominal
118	EiC_CLS_DESC					None	None	50	Left	Nominal
119	EiC_ACTION_ZONE					None	None	8	Right	Scale
120	EiC_ACTION_ZONE_D		255	0		None	None	50	Left	Nominal
121	FRESH_START		8	2		None	None	8	Right	Nominal
122	FRESH_START_DESC		255	0		None	None	50	Left	Nominal
123	EARLY_EX		8	2		None	None	8	Right	Nominal
124	EARLY_EX_DESC		255	0		None	None	50	Left	Nominal
125	SIXTH_FORM		8	2		None	None	8	Right	Nominal
126	SIXTH_FORM_DESC		255	0		None	None	50	Left	Nominal
127	EY	Numeric	8	2		None	None	8	Right	Nominal
128	EY_DESC	String	255	0		None	None	50	Left	Nominal
129	PT_FLAG	String	255	0		None	None	50	Left	Nominal
130	ASC_LOW_AGE	Numeric	8	2		None	None	8	Right	Nominal
131	ASC_HIGH_AGE	Numeric	8	2		None	None	8	Right	Nominal
132	APP_SPEC_BOARDER_PUPIL	String	255	0		None	None	50	Left	Nominal
133	coded2007schlnamea	Numeric	5	0		{1, # Due to O	None	20	Right	Nominal
134	toe06codeda	Numeric	2	0		{1, Academies	None	13	Right	Nominal
135	SLEA2007	Numeric	3	0		{1, Barking an	None	10	Right	Nominal
136	SGOR07a	Numeric	2	0		{1, North East}	None	8	Right	Nominal
137	sward06	Numeric	4	0		{2, Abbey}...	None	9	Right	Nominal
138	sdistrict06	Numeric	3	0		{2, Aberdeen C	None	13	Right	Nominal
139	sconstituency06	Numeric	3	0		{2, Aberavon}..	None	17	Right	Nominal
140	sdiocese06	Numeric	2	0		{2, [Not Applic	None	12	Right	Nominal
141	sselctive06	Numeric	1	0		{2, [Not Applic	None	13	Right	Nominal
142	sgender06	Numeric	1	0		{2, [Not Applic	None	11	Right	Nominal
143	stoe06	Numeric	2	0		{1, Academies	None	8	Right	Nominal
144	spoe06	Numeric	1	0		{1, [Not Applic	None	8	Right	Nominal
145	sdenom06	Numeric	2	0		{2, Buddhist}...	None	10	Right	Nominal
146	surbanrural06	Numeric	2	0		{2, Hamlet and	None	15	Right	Nominal

In the case of the English Pupil Dataset, and as shown in Figure 55, this prompts the message that 'The Customs Tables dialogue cannot be opened if any variable has more than 12000 labels'. The English Pupil Dataset easily breaches that limit. A Web search showed this to be a 'known issue' with SPSS version 14, and if you encounter it you may well need specialist IT help. As an interim solution in City Hall, and working with the SPSS helpdesk, an older version of the SPSS Tables module was 'reactivated', and used to run Tables through the 'General Tables' option.

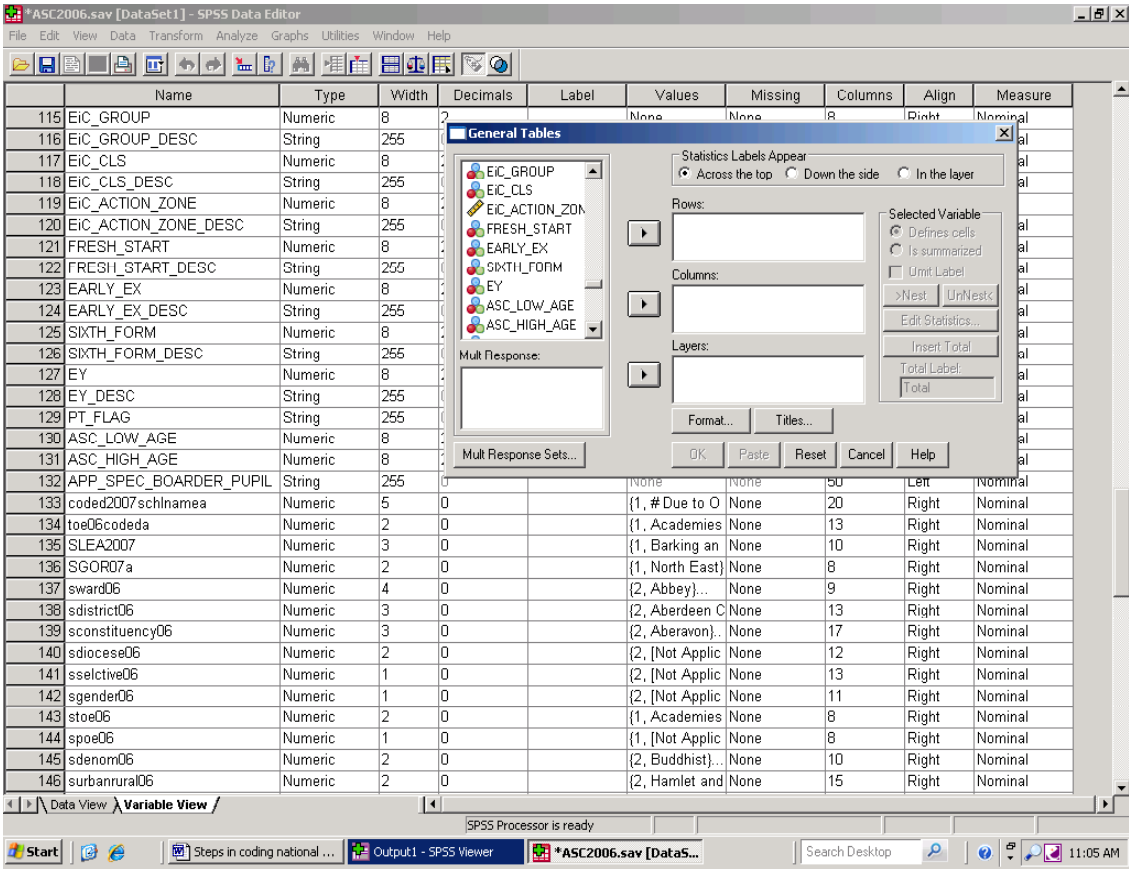
The General Tables option allows the user to select variables for display in rows and columns as in crosstabs, and allows for the inclusion of totals and for variables to be 'nested' within other variables. However, it will not tabulate long string variables.

Figure 55. An issue with Custom Tables in SPSS Version 14



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
115	EiC_GROUP	Numeric	8	2		None	None	8	Right	Nominal
116	EiC_GROUP_DESC	String	255	0		None	None	50	Left	Nominal
117	EiC_CLS	Numeric	8	2		None	None	8	Right	Nominal
118	EiC_CLS_DESC	String	255	0		None	None	50	Left	Nominal
119	EiC_ACTION_ZONE	Numeric	8	2		None	None	8	Right	Scale
120	EiC_ACTION_ZONE_DESC	String	255	0		None	None	50	Left	Nominal
121	FRESH_START	Numeric	8	2		None	None	8	Right	Nominal
122	FRESH_START_DESC	String	255	0		None	None	50	Left	Nominal
123	EARLY_EX	Numeric	8	2		None	None	8	Right	Nominal
124	EARLY_EX_DESC	String	255	0		None	None	50	Left	Nominal
125	SIXTH_FORM	Numeric	8	2		None	None	8	Right	Nominal
126	SIXTH_FORM_DESC	String	255	0		None	None	50	Left	Nominal
127	EY	Nur						8	Right	Nominal
128	EY_DESC	Stri						50	Left	Nominal
129	PT_FLAG	Stri						50	Left	Nominal
130	ASC_LOW_AGE	Nur						8	Right	Nominal
131	ASC_HIGH_AGE	Nur						8	Right	Nominal
132	APP_SPEC_BOARDER_PUPIL	String	255	0		None	None	50	Left	Nominal
133	coded2007schlnamea	Numeric	5	0		{1, # Due to O	None	20	Right	Nominal
134	toe06codeda	Numeric	2	0		{1, Academies	None	13	Right	Nominal
135	SLEA2007	Numeric	3	0		{1, Barking an	None	10	Right	Nominal
136	SGOR07a	Numeric	2	0		{1, North East}	None	8	Right	Nominal
137	sward06	Numeric	4	0		{2, Abbey}...	None	9	Right	Nominal
138	sdistrict06	Numeric	3	0		{2, Aberdeen C	None	13	Right	Nominal
139	sconstituency06	Numeric	3	0		{2, Aberavon}..	None	17	Right	Nominal
140	sdiocese06	Numeric	2	0		{2, [Not Applic	None	12	Right	Nominal
141	sselctive06	Numeric	1	0		{2, [Not Applic	None	13	Right	Nominal
142	sgender06	Numeric	1	0		{2, [Not Applic	None	11	Right	Nominal
143	stoe06	Numeric	2	0		{1, Academies	None	8	Right	Nominal
144	spoe06	Numeric	1	0		{1, [Not Applic	None	8	Right	Nominal
145	sdenom06	Numeric	2	0		{2, Buddhist}...	None	10	Right	Nominal
146	surbanrural06	Numeric	2	0		{2, Hamlet and	None	15	Right	Nominal

Figure 56. SPSS General Tables, Columns, Rows and Totals



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
115	EiC_GROUP	Numeric	8	2		None	None	8	Right	Nominal
116	EiC_GROUP_DESC	String	255							al
117	EiC_CLS	Numeric	8							al
118	EiC_CLS_DESC	String	255							al
119	EiC_ACTION_ZONE	Numeric	8							
120	EiC_ACTION_ZONE_DESC	String	255							al
121	FRESH_START	Numeric	8							al
122	FRESH_START_DESC	String	255							al
123	EARLY_EX	Numeric	8							al
124	EARLY_EX_DESC	String	255							al
125	SIXTH_FORM	Numeric	8							al
126	SIXTH_FORM_DESC	String	255							al
127	EY	Numeric	8							al
128	EY_DESC	String	255							al
129	PT_FLAG	String	255							al
130	ASC_LOW_AGE	Numeric	8							al
131	ASC_HIGH_AGE	Numeric	8							al
132	APP_SPEC_BOARDER_PUPIL	String	255	0		None	None	50	Left	Nominal
133	coded2007schlnamea	Numeric	5	0		{1, # Due to O	None	20	Right	Nominal
134	toe06codeda	Numeric	2	0		{1, Academies	None	13	Right	Nominal
135	SLEA2007	Numeric	3	0		{1, Barking an	None	10	Right	Nominal
136	SGOR07a	Numeric	2	0		{1, North East}	None	8	Right	Nominal
137	sward06	Numeric	4	0		{2, Abbey}...	None	9	Right	Nominal
138	sdistrict06	Numeric	3	0		{2, Aberdeen C	None	13	Right	Nominal
139	sconstituency06	Numeric	3	0		{2, Aberavon}..	None	17	Right	Nominal
140	sdiocese06	Numeric	2	0		{2, [Not Applic	None	12	Right	Nominal
141	sselctive06	Numeric	1	0		{2, [Not Applic	None	13	Right	Nominal
142	sgender06	Numeric	1	0		{2, [Not Applic	None	11	Right	Nominal
143	stoe06	Numeric	2	0		{1, Academies	None	8	Right	Nominal
144	spoe06	Numeric	1	0		{1, [Not Applic	None	8	Right	Nominal
145	sdenom06	Numeric	2	0		{2, Buddhist}...	None	10	Right	Nominal
146	surbanrural06	Numeric	2	0		{2, Hamlet and	None	15	Right	Nominal

Figure 56 illustrates this point. The SPSS 'Variable View' of the 2006 SC dataset shows variables 115 to 146. It also shows the General Tables Windows view of variables. Amongst others, variables 116, 118, 120, 122 and 124 are not listed. These are long string variables, and long string variables are not accommodated in SPSS Tables. As with a number of other string variables in the NPD extracts, numeric, labelled, equivalents of these excluded variables have been created. Once quality checks have been run, the source string versions will eventually be deleted from the working file to help keep file size within limits.

Section 13 used the string variable containing information on pupil home ward to illustrate the Autorecode procedure, and made the point that wards with exactly the same name would be given the same code number. Figure 58 illustrates what at first glance is a useful Table for checking the number of pupils living in individual wards. The Row pane in the dialogue box shows that the autorecoded ward names have been grouped under a variable 'dmag1'. That variable identifies individual London boroughs, and other LEAs around London, and groups other wards in terms of their region. The variable name 'dmag1' was selected from the variable list on the left of the General Tables dialogue box, and transferred to the 'Rows' pane.

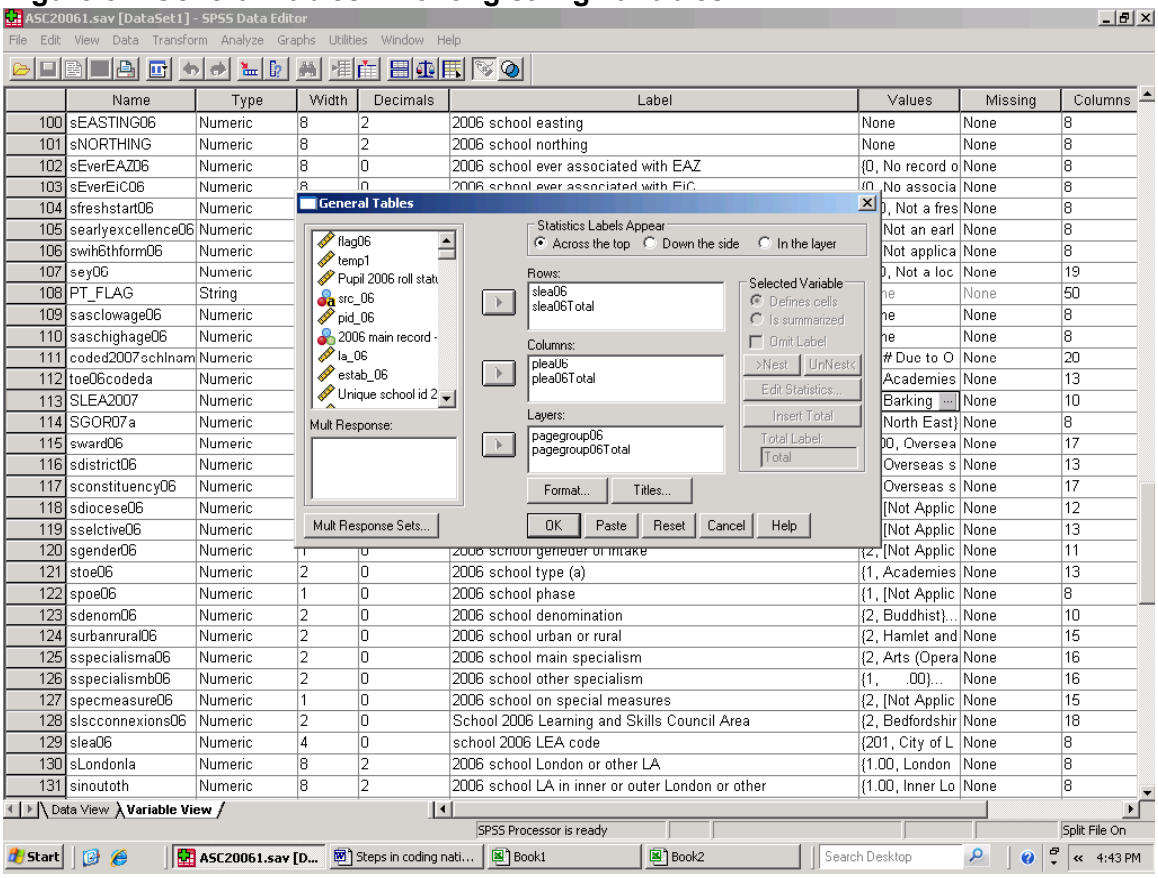
Following this, the autorecoded home ward was selected in the same way, and the name transferred to the Row pane. Highlighting the autorecode home ward variable in the 'Rows' section, and then selecting 'Nest' will group pupil home ward within pupil home local authority area. Where there is more than one ward with the same name, these will now be split between the appropriate local authorities.

The Columns section in the dialogue box contains the variable 'sdmag1', which groups the schools attended in terms of their maintaining local authority.

Figure 58 shows that output is to be layered by school region, GOR being the acronym for Government Office for the Region. Tables will be given separately for each English region in layers, as in a sponge cake. Output of this type, grouping schools by phase, is shown in Figure 64.

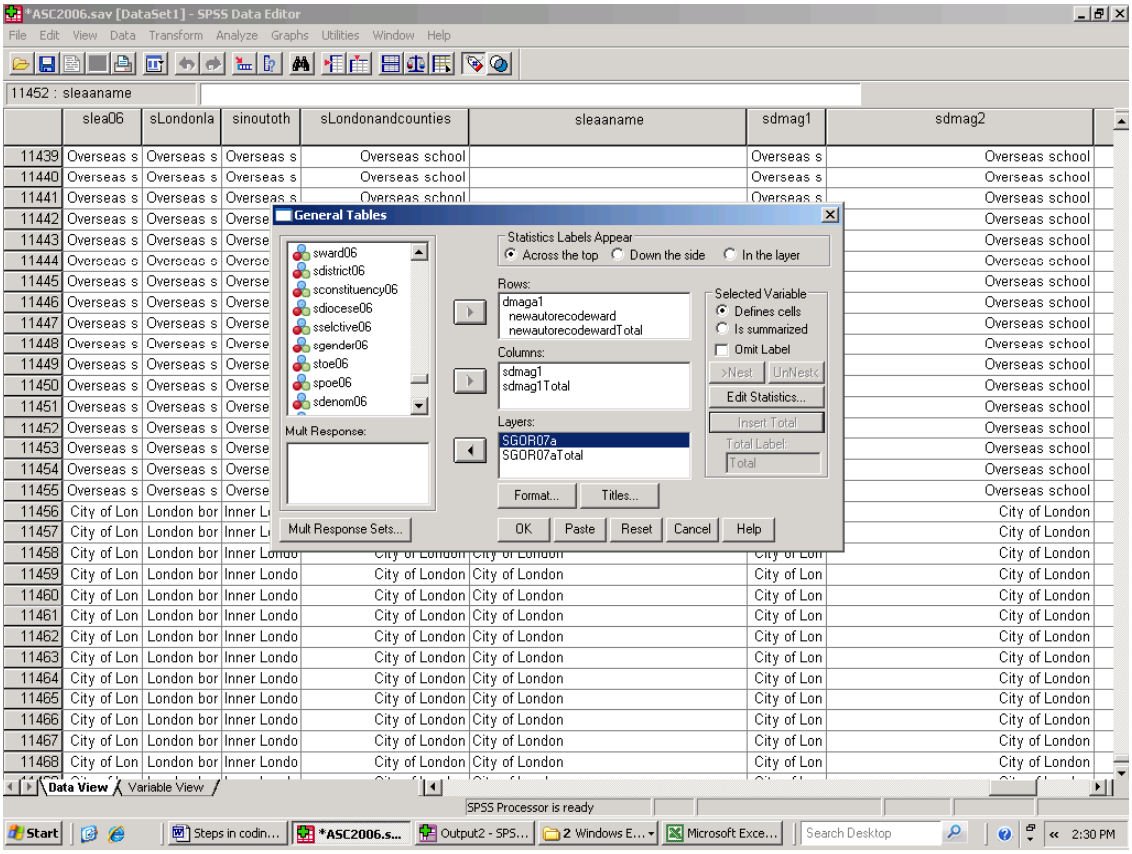
A Table for the first region listed in the SGOR07a variable will be shown first. A drop down list of regions can be selected in the 'Layers' section of the Table, and another region can be shown.

Figure 57. General Tables – no long string variables



	Name	Type	Width	Decimals	Label	Values	Missing	Columns
100	sEASTING06	Numeric	8	2	2006 school easting	None	None	8
101	sNORTHING	Numeric	8	2	2006 school northing	None	None	8
102	sEverEAZ06	Numeric	8	0	2006 school ever associated with EAZ	{0, No record o	None	8
103	sEverEiC06	Numeric	8	0	2006 school ever associated with EiC	{0, No associa	None	8
104	sfreshstart06	Numeric				, Not a fres	None	8
105	searlyexcellence06	Numeric				Not an earl	None	8
106	swih6thform06	Numeric				Not applica	None	8
107	sey06	Numeric				, Not a loc	None	19
108	PT_FLAG	String					None	50
109	sasclowage06	Numeric					None	8
110	saschighage06	Numeric					None	8
111	coded2007schlnam	Numeric				# Due to O	None	20
112	toe06codeda	Numeric				Academies	None	13
113	SLEA2007	Numeric				Barking	None	10
114	SGOR07a	Numeric				North East}	None	8
115	sward06	Numeric				00, Oversea	None	17
116	sdistrict06	Numeric				Overseas s	None	13
117	sconstituency06	Numeric				Overseas s	None	17
118	sdiocese06	Numeric				[Not Applic	None	12
119	sselctive06	Numeric				[Not Applic	None	13
120	sgender06	Numeric	1	0	2006 school gender of intake	{2, [Not Applic	None	11
121	stoe06	Numeric	2	0	2006 school type (a)	{1, Academies	None	13
122	spoe06	Numeric	1	0	2006 school phase	{1, [Not Applic	None	8
123	sdenom06	Numeric	2	0	2006 school denomination	{2, Buddhist}...	None	10
124	surbanrural06	Numeric	2	0	2006 school urban or rural	{2, Hamlet and	None	15
125	sspecialisma06	Numeric	2	0	2006 school main specialism	{2, Arts (Opera	None	16
126	sspecialismb06	Numeric	2	0	2006 school other specialism	{1, .00}...	None	16
127	specmeasure06	Numeric	1	0	2006 school on special measures	{2, [Not Applic	None	15
128	slsccconnexions06	Numeric	2	0	School 2006 Learning and Skills Council Area	{2, Bedfordshir	None	18
129	slea06	Numeric	4	0	school 2006 LEA code	{201, City of L	None	8
130	sLondonla	Numeric	8	2	2006 school London or other LA	{1.00, London	None	8
131	sinoutoth	Numeric	8	2	2006 school LA in inner or outer London or other	{1.00, Inner Lo	None	8

Figure 58. Organising Tables in Layers



In practice the Table set out in Figure 58 would be extremely large; invoking that Table with a 7.5 million case dataset brings SPSS to a standstill. If a Table of that sort is run, and SPSS remains locked, there may be no alternative to using Windows task manager to end the task. Unfortunately this would shut SPSS down, and any unsaved data would be lost.

As a rule of thumb, avoid running complex Tables procedures with large datasets. This Section provides some advice on how to avoid this problem, and further advice is available in Section 18 on splitting datasets and layering tables.

There are precautionary measures that can be taken when using what is essentially a pc version of SPSS to analyse large datasets. As a first measure, save the dataset before running large, complex Tables. This means that if SPSS 'locks' when faced with an overly large table, Task Manager can be used to close SPSS down without work being lost. A further, and less crisis oriented option is to work with subsets of data, using the 'Select Cases' procedures to select, for example, pupils living in a particular region or pupils of a particular age.

If you find that running an SPSS Table is close to the limits of your computer's capacity, do not use 'multiple nesting'. In the case shown above, nesting the additional variable 'pupil home census output area' under the autorecoded ward variable would be a case of multiple nesting. SPSS Tables may be produced, but there is a risk that their content would be garbled. SPSS Tables can provide a useful way of checking data quality, but it does have its limitations.

18. Organising data in Tables for other users, and using 'Split file' 'Select Data' and 'Layer Table' to minimise the risk of overload

Section 16 referred to one project using pupil level data

In 2009, one project using pupil level data required, for the first time, the identification of the local authority district in which each school in England was located.

On March 3rd 2009 Parliament debated the shortage of places for 1st year pupils in primary schools, and there were subsequent questions put to Ministers on that issue in the House of Commons. On April 8th 2009, *The Daily Telegraph* Newspaper carried a front-page story on a shortfall in the money needed to meet the demand for VIth form places in English maintained (state) secondary schools. While this was not an exclusively London issue, some London boroughs experienced a marked rise in demand which could not be accommodated easily within existing provision.

The GLA provides projections of demand for school places in 25 of London's 33 local (education) authorities, and the production of a wider, pan-London view of demand for school places had already been discussed with those boroughs. A short DMAG Briefing (2005 – 36), gave an early view of options involved, and a pilot was developed incorporating a view for each London borough, and for the counties and unitary authorities around London.

The SPSS Tables procedure can use pupil level data to generate roll summaries for individual schools. In this instance pupil headcounts by gender and single years of age in maintained schools for pupils aged 4 to 19, and distinguishing between primary and secondary schools were required. However, the number of schools involved was equivalent to the number in a medium-sized European state and, as Section 17 shows, there are problems in producing very large Tables in SPSS. Work with roll data is used here to illustrate ways around those restrictions. A further point is that when we ready data for analysis, we need to take account of what those analytical needs area. A well-intended guess may not be enough.

The exercise is broken down into smaller steps to reduce the risk of overloading the SPSS Tables procedure. For example, based on information added from EduBase, a 'sprisecspec' variable has been created which shows whether a pupil attended a nursery, primary, secondary or special school. (Pupils in mainstream schools such as middle schools and Academies, which cater for pupils in more than one phase, were subsequently reallocated to the primary or secondary record depending on their age). The 'Select if' facility was used so that Tables were produced for each type of school separately.

Additionally, in the same way that a 'flag' variable of 1 had been created for each year of pupil data, as in the creation of 'flag06' for the 2006 English National Pupil Dataset (see section 6) a flag variable had been created for those pupils attending schools maintained by London local authorities or by local (education) authorities around London. Selecting for type of school and for local authorities in and around London reduces the volume of data SPSS works with, and produces output which is closer to the user's requirements than would otherwise be the case.

The 'Split File' procedure shown in Figure 60 is a further way of breaking down the exercise into more manageable chunks. This is accessed by selecting 'Data' from the SPSS main menu, then by selecting 'Split File' from the resulting dropdown list. This allows data to be analysed in groups, by individual local (education) authorities. These have already been coded so that the order in which they appear reflects user needs. Additionally, as noted and as Figure 59 shows, the 'Select if' facility has already been used to restrict the analysis to London and the neighbouring shire counties and unitary authorities.

The next step aimed to produce a Table which matched user requirements (i.e. by gender and single years of age, but also with school name and DCSF number). Figure 61 shows the General Tables window with schools nested in administrative districts in rows, and with single year of age nested within gender in columns. Figure 62 shows specimen output from this procedure.

The Figure makes two points. Firstly, the procedure has run successfully. Breaking the exercise down into smaller parts has worked. The second point, is to that a question mark hangs over the output produced.

Figure 59. Using 'Select Cases' to avoid overburdening 'Tables' when working with large datasets

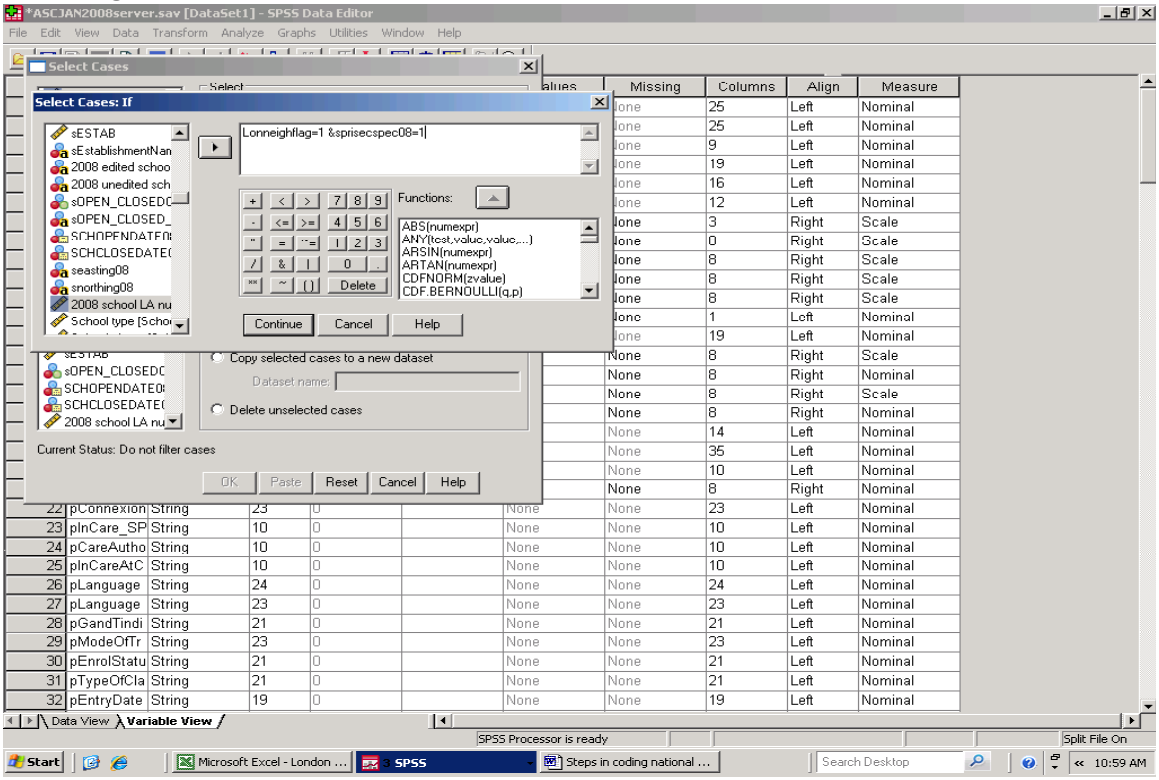


Figure 60. Using 'Split File' to avoid overburdening 'Tables' when working with large datasets

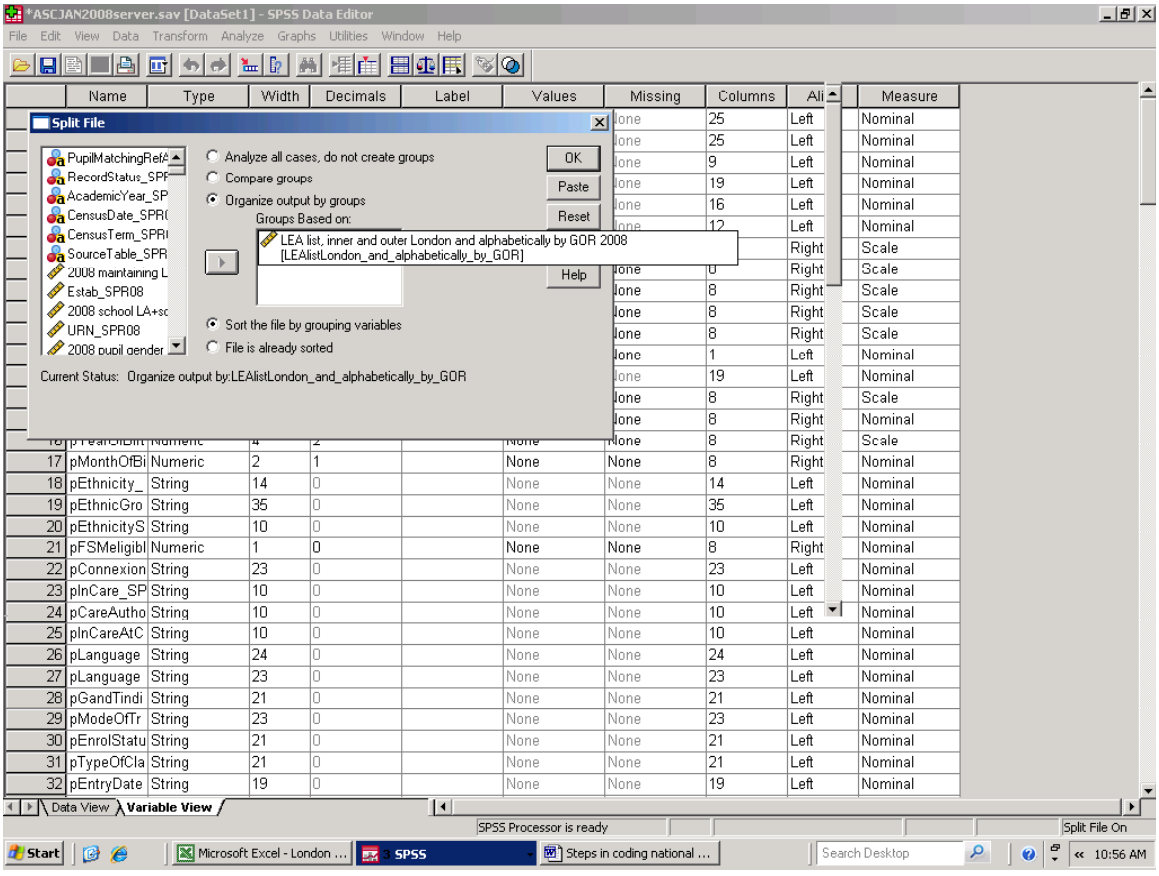


Figure 61. Organising data in a Table to meet user requirements?

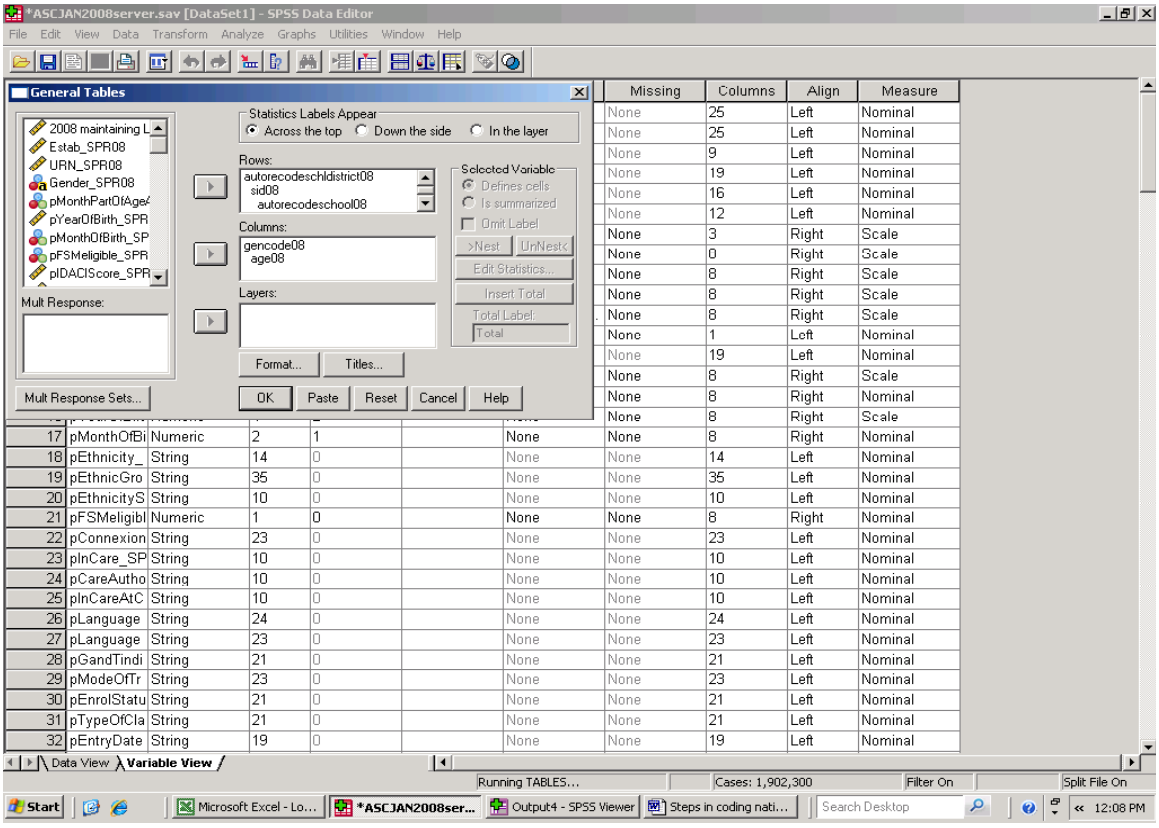


Figure 62. Output from the Table. Too much background noise?

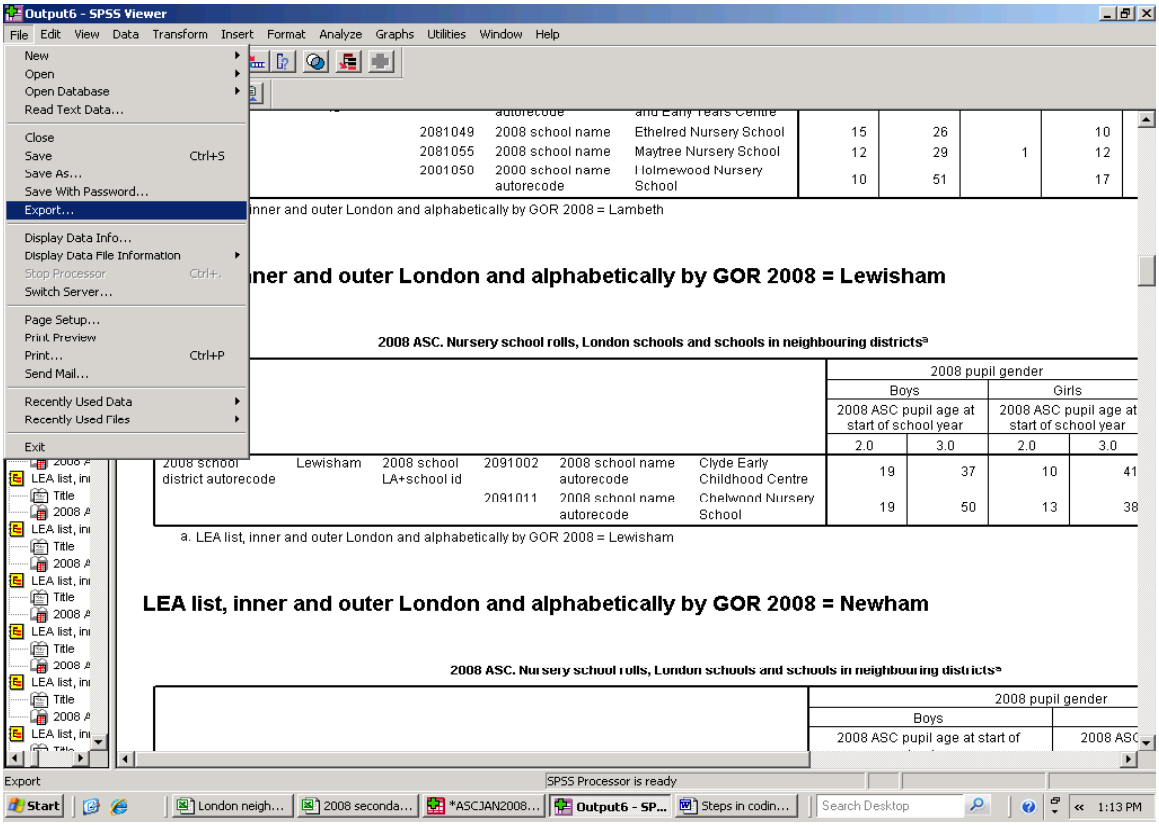


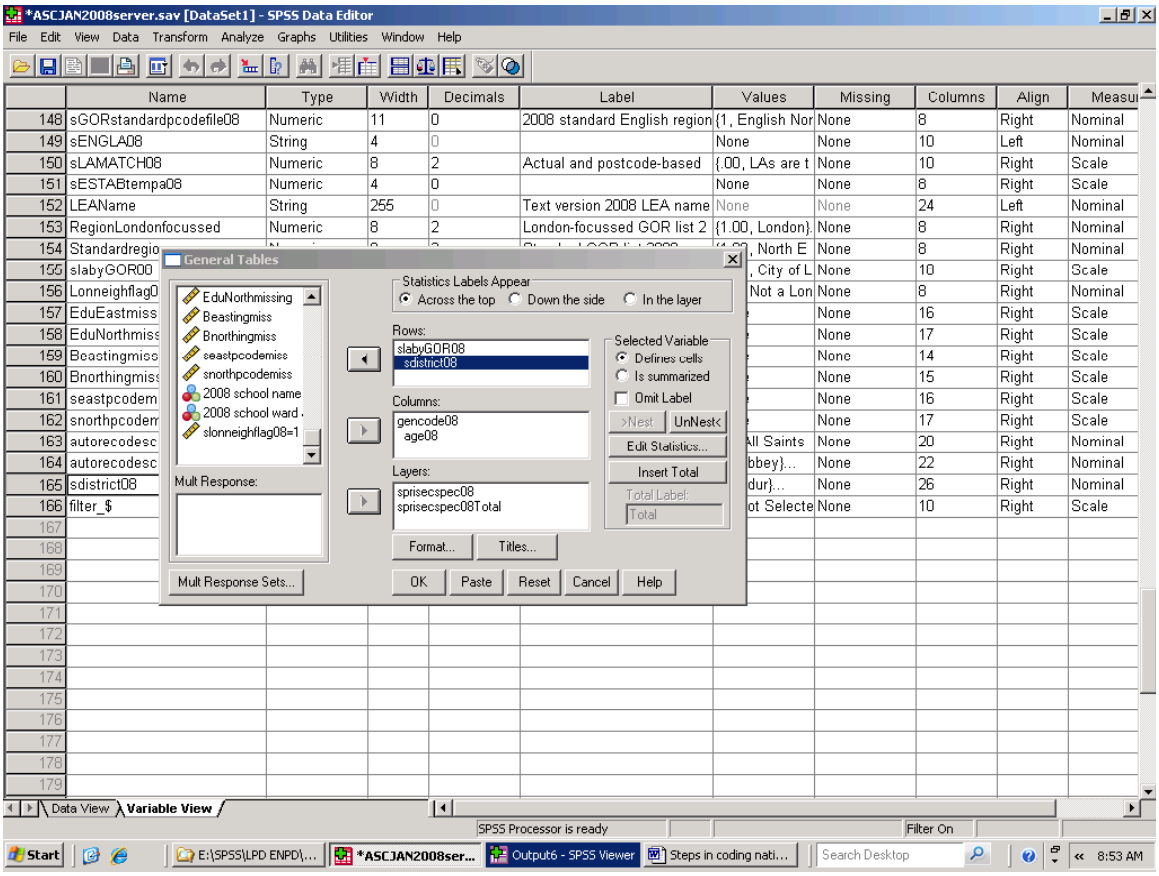
Figure 62 shows school rolls in individual institutions grouped within districts, by gender and single years of age. The Tables shown are all from local authority nursery schools, that is Tables are organised into separate educational phases. That grouping will also provide separate data for primary and secondary schools, as required. Output is also split by maintaining local authority, which means that there is a separate Table for each local authority and the schools it maintains. Each Table is also clearly labelled, and there should be no doubt about which variables have been used. As it stands, the output *may* meet the user's needs, but equally it may not.

SPSS datasets are single flat files with data in columns and rows. If the user needs output in that type of format then the output in Figure 62 has a great deal of 'redundant' space between the Tables that will need to be edited out. The layering facility within the Tables module avoids that issue.

In this example, the 2008 English Pupil Dataset is used and pupil records in London and the surrounding shire counties and unitary authority have been selected as previously. The Table will again show pupils by gender and single years of age, and the district of the school attended (but not in this case the name of the school attended, though that could be included.).

The dataset is split on the 'slabyGOR' as shown in Figure 60, but the variable dealing with school phase (sprisecspec08) is not included in the 'Select Cases exercise as it was in the exercise shown in Figure 59. Figure 63 shows that, on this occasion, the school phase variable is included in the Layer section of the Tables Window.

Figure 63. Using Layers in SPSS Tables to organise data



Placing the variable sprisecspec08 in the layer box, as shown in Figure 63, means that output is given separately for local authority nursery schools, primary schools, secondary schools including Academies and CTCs, and special schools. (Hopefully, it will go without saying that output can only be layered in this way if the variable sprisecspec08 exists in the first place and is suitably labelled.) Part of the output from this procedure is shown in Figure 64, with a dropdown box which allows the reader to select the type of school required.

Figure 64. Layered output in SPSS

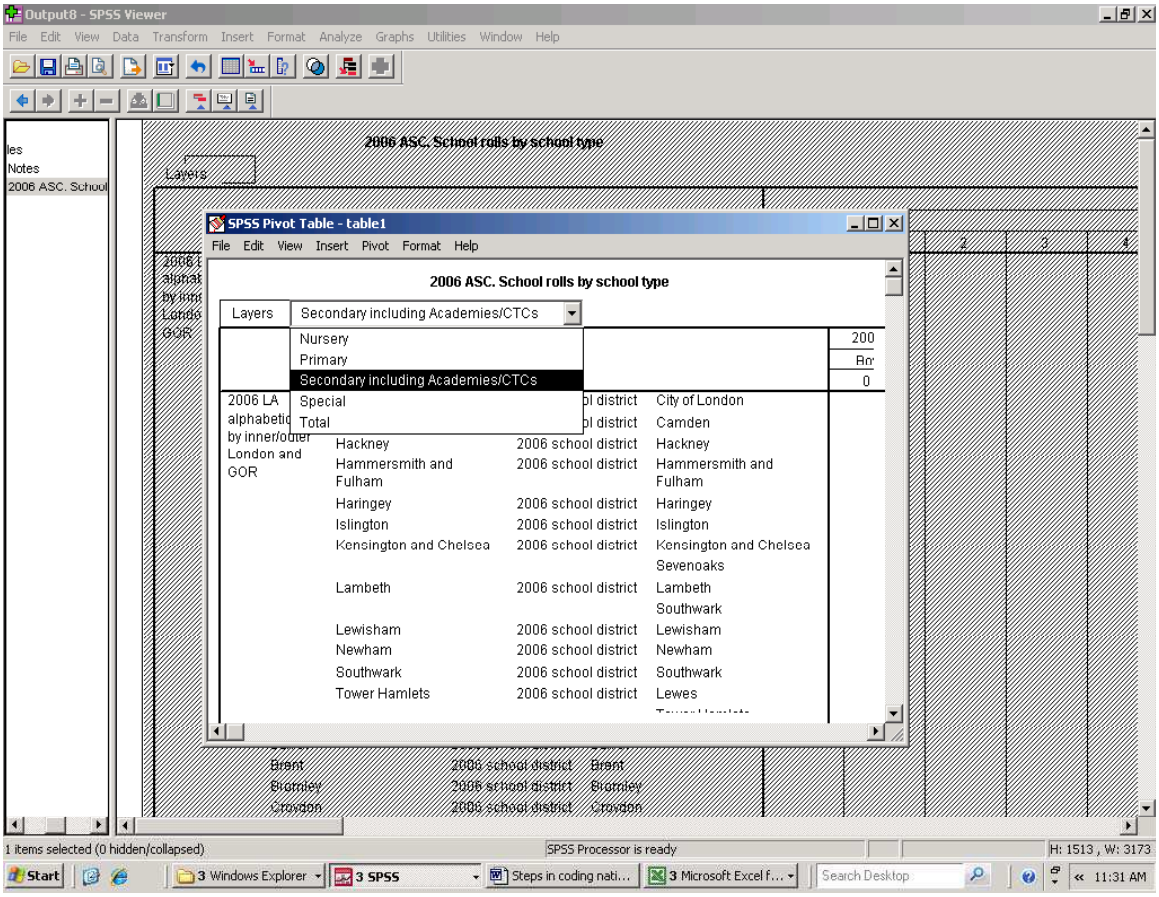


Figure 65. School rolls by district, gender and single years of age. Less background noise

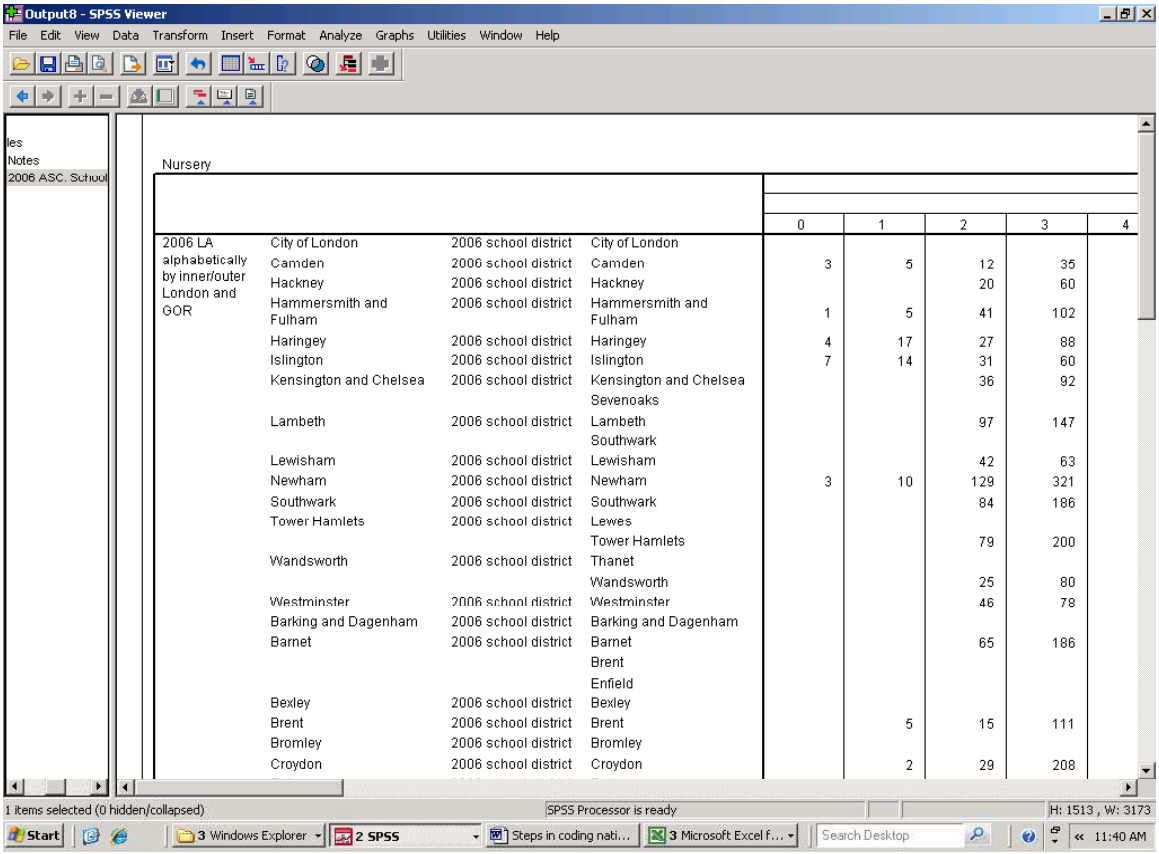
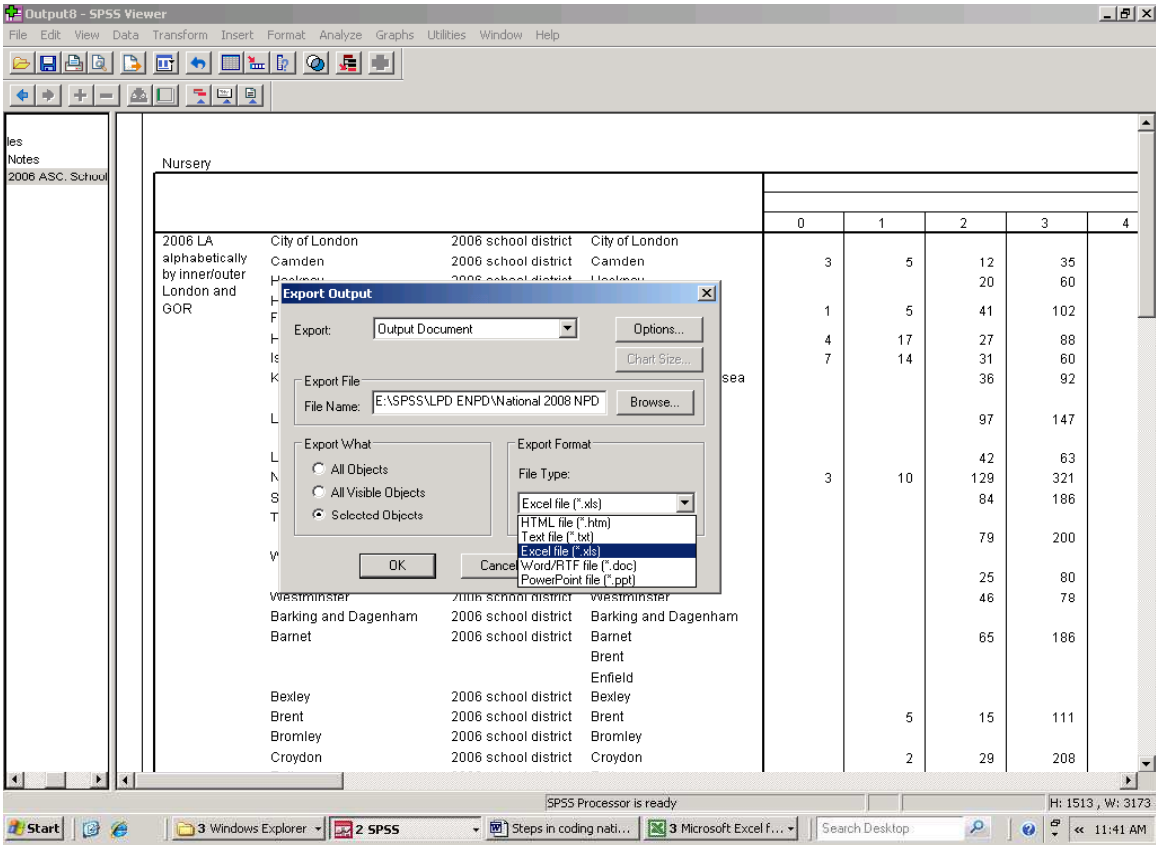


Figure 66. Exporting Tables Output to EXCEL



The pilot pan-London roll projection project is carried out in EXCEL, though the structure of data file is close to, if not identical with, an SPSS dataset. To export an SPSS Table to EXCEL, select 'File' in the main SPSS menu, and then select 'Export' in the resulting drop down list. This produces an 'Export Output' dialogue box, shown in Figure 66. This enables you to select the type of output from a list of choices, and provides a 'Browse' facility which allows you to decide where the file is to go. Figure 67 shows part of the EXCEL file that has been created. The file contains four Tables, one for each type of school in line with the codes in sprisecspec06. Each of the four layers has been successfully exported to a single EXCEL worksheet.

Left clicking on the output in SPSS produces a pivot table, with an 'arrow' in the upper left hand part of the table. Selecting this produces a drop down list, showing the four types of school that have been coded (hence the word 'layer' – each table is positioned within a layer). Tables can also be copied one layer at a time into EXCEL working within the Pivot Table. With the Pivot Table open, select 'Edit', then 'Select' and 'Table', and then 'Edit' and 'Copy'. This allows the layer in view to be copied and, in principle all layers can be copied across to EXCEL in turn in this way. (Note that the full set of layers cannot be copied in a single 'Export' exercise from within a pivot table. Selecting 'File' from within the Pivot Table window gives only one option – to close the window.)

The Tables produced using the 'layer' facility do not have text between the rows for each case (in this instance local education authorities) and this, depending on what is needed, can be an advantage. Additionally each Table contains the same rows and the same columns, which has its advantages in this instance since the need is for a single block of data with the same age group shown in the same column.

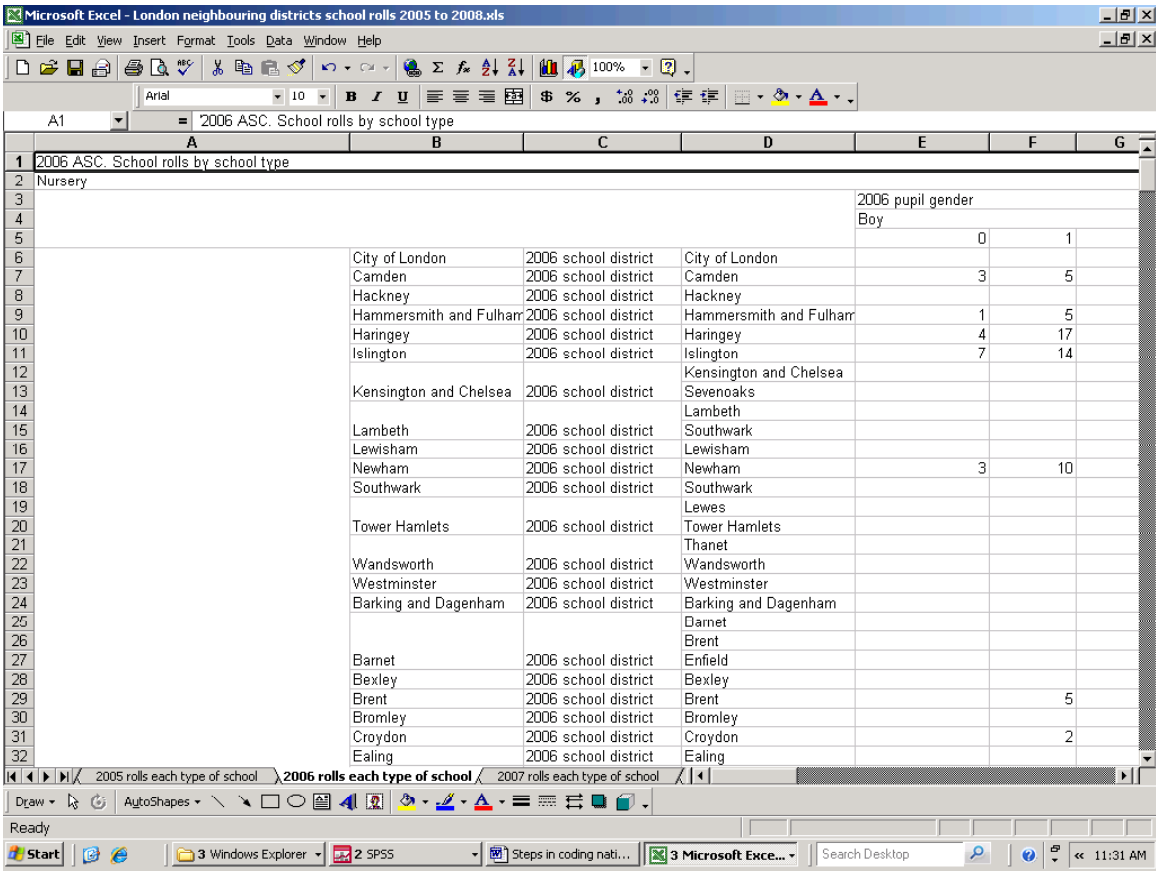
By contrast, the procedure that produced the output shown in Figure 62, provides the number in an age group only when there is one or more pupil in that age group. The youngest age group in a nursery school can be aged one to five, The first age group column in an individual LA primary phase Table, of the type shown in Figure 62, can be any one of those age groups,

The corollary is that, given the approach shown in Figure 65, if there are pupils listed in an age group or category in one Table, then that age group or category will appear in all Tables, irrespective of whether there are any pupils are recorded in that phase. Kensington and Chelsea, for example, maintains a special school in Sevenoaks in Kent. Figure 65 might appear to be suggesting that the (Royal) Borough of Kensington and Chelsea also maintains a nursery school in Sevenoaks, (and later sections also appear to suggest that it maintains a primary and secondary school in that town). For Kensington

and Chelsea, the Sevenoaks rows are blank in all sections other than the special school section.

Deleting redundant rows, columns, or text between Tables is a chore, but there is scope for keeping it in check if the Tables are carefully designed, and if the choice between splitting a file and using the layering facility is thought through in advance. More to the point, different users have different needs. The choice of Table can advance or hold back further analysis and that choice needs to be made with care.

Figure 67. The Table in EXCEL



	A	B	C	D	E	F	G
1	2006 ASC. School rolls by school type						
2	Nursery						
3					2006 pupil gender		
4					Boy		
5					0	1	
6		City of London	2006 school district	City of London			
7		Camden	2006 school district	Camden	3	5	
8		Hackney	2006 school district	Hackney			
9		Hammersmith and Fulham	2006 school district	Hammersmith and Fulham	1	5	
10		Haringey	2006 school district	Haringey	4	17	
11		Islington	2006 school district	Islington	7	14	
12				Kensington and Chelsea			
13		Kensington and Chelsea	2006 school district	Sevenoaks			
14				Lambeth			
15		Lambeth	2006 school district	Southwark			
16		Lewisham	2006 school district	Lewisham			
17		Newham	2006 school district	Newham	3	10	
18		Southwark	2006 school district	Southwark			
19				Lewes			
20		Tower Hamlets	2006 school district	Tower Hamlets			
21				Thanet			
22		Wandsworth	2006 school district	Wandsworth			
23		Westminster	2006 school district	Westminster			
24		Barking and Dagenham	2006 school district	Barking and Dagenham			
25				Darnet			
26				Brent			
27		Barnet	2006 school district	Enfield			
28		Bexley	2006 school district	Bexley			
29		Brent	2006 school district	Brent		5	
30		Bromley	2006 school district	Bromley			
31		Croydon	2006 school district	Croydon		2	
32		Ealing	2006 school district	Ealing			

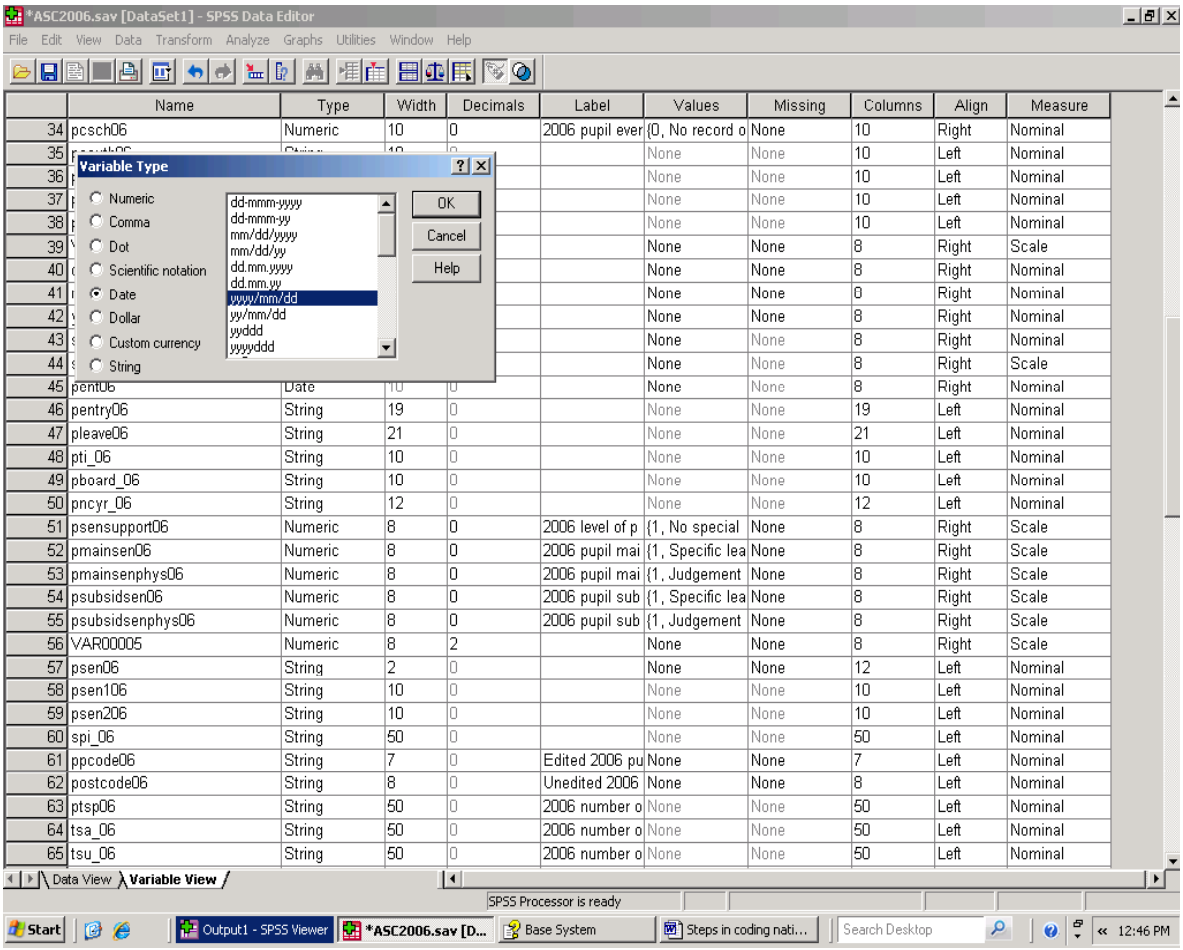
19. Working with dates

SPSS holds, out of sight, the time *in seconds* from 14th October 1582 to the date shown in a date variable. The 14th October 1582 is taken as the standardised date when the reformed, Gregorian, calendar was introduced to Europe. (This replaced the earlier Julian calendar, named after Julius Caesar, which had been in use since 46 BC and contained 'cumulative' errors. Pope Gregory agreed to the reformed calendar in 1582, hence the name the Gregorian calendar. Britain adopted the Gregorian calendar in 1752.)

Date variables can be used to calculate the passage of time, as in calculating pupil age from data of birth, or some other passage/length of time using other dates. This Section introduces that work, including reworking string data to create a date, comparing date of birth and admission date to calculate pupil age when admitted to the current school, and creating a 'dummy' admission date which can be used to calculate how old each pupil would have been in whole years at the start of the school year in which each pupil was first admitted to his or her current school.

As a note of caution, this involves changing variables from a string to a numeric format, and back again, and data can be wiped out in the process. Use the SPSS Data View window to look at the data when you have changed its type. If data has been wiped out, simply left click on the 'Undo' button at the top of the window and this will restore the data *as long as you have not saved the file after changing the character of variable and before looking at the data.* If a variable's character is to be altered on several occasions, copying that variable to a new variable offers further protection (you may lose data, but at least your source information will still be there). As a more general principle, if you are about to carry out extensive work where many of the steps are new to you, take a backup copy of the dataset as a whole.

Figure 68. Choosing date formats



	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
34	pcsch06	Numeric	10	0	2006 pupil ever	{0, No record o	None	10	Right	Nominal
35						None	None	10	Left	Nominal
36						None	None	10	Left	Nominal
37						None	None	10	Left	Nominal
38						None	None	10	Left	Nominal
39						None	None	8	Right	Scale
40						None	None	8	Right	Nominal
41						None	None	0	Right	Nominal
42						None	None	8	Right	Nominal
43						None	None	8	Right	Nominal
44						None	None	8	Right	Scale
45	pentU6	Date	10	0		None	None	8	Right	Nominal
46	pentry06	String	19	0		None	None	19	Left	Nominal
47	pleave06	String	21	0		None	None	21	Left	Nominal
48	pti_06	String	10	0		None	None	10	Left	Nominal
49	pboard_06	String	10	0		None	None	10	Left	Nominal
50	pncyr_06	String	12	0		None	None	12	Left	Nominal
51	psensupport06	Numeric	8	0	2006 level of p	{1, No special	None	8	Right	Scale
52	pmainsen06	Numeric	8	0	2006 pupil mai	{1, Specific lea	None	8	Right	Scale
53	pmainsenphys06	Numeric	8	0	2006 pupil mai	{1, Judgement	None	8	Right	Scale
54	psubsidsen06	Numeric	8	0	2006 pupil sub	{1, Specific lea	None	8	Right	Scale
55	psubsidsenphys06	Numeric	8	0	2006 pupil sub	{1, Judgement	None	8	Right	Scale
56	VAR00005	Numeric	8	2		None	None	8	Right	Scale
57	psen06	String	2	0		None	None	12	Left	Nominal
58	psen106	String	10	0		None	None	10	Left	Nominal
59	psen206	String	10	0		None	None	10	Left	Nominal
60	spi_06	String	50	0		None	None	50	Left	Nominal
61	ppcode06	String	7	0	Edited 2006 pu	None	None	7	Left	Nominal
62	postcode06	String	8	0	Unedited 2006	None	None	8	Left	Nominal
63	ptsp06	String	50	0	2006 number o	None	None	50	Left	Nominal
64	tsa_06	String	50	0	2006 number o	None	None	50	Left	Nominal
65	tsu_06	String	50	0	2006 number o	None	None	50	Left	Nominal

The NPD contains a record of the date a pupil was admitted to his or her current school. The format in the text file is YYYY-MM-DD 00:00:00 where, YYYY is the year, MM is the number of the month and DD is the number of the day of the month. The zeros at the right hand end are redundant, and this variable cannot be read by SPSS as a date in this format.

Take the variable into SPSS as a string variable, and use the substring facility to

Compute padmit06 =substr(sourcevariable,1,10).

Then in SPSS Variable View change that variable from a string variable to a date variable by left clicking on 'Type' cell for padmit06, and selecting 'Date' and mm/dd/yyyy format in the Variable Type window. (See Figure 68).

In some instances, you may not be able to change a string variable containing date information in this way, and assume that is the case here. A date can be constructed using the component parts of the NPD admission date. It is a slightly longer process, but it works and is useful to know. Assume that you are dealing with the date format given in text on the previous page. Create three separate *string* variables dd, mm, and yyyy with widths of 2, 2 and 4 respectively. For the new dd variable

Compute dd=substr(NPDadmissiondate,9,2).

You have asked SPSS to look at the source variable, and to put two characters starting at character number 9 in your new 'dd' variable. For the 'mm' variable

Compute mm=substr(NPDadmissiondate,6,2)

and for the 'yyyy' variable

Compute yyyy=substr(NPDadmissiondate,1,4).

In what follows, the procedures shown will not work if the three 'new' variables are not changed from string to numeric variables. In the SPSS Variable View window, change dd, mm and yyyy to *numeric* variables by left clicking on the 'Type' cell for each variable and making the necessary changes. Give the variables a width of 2, 2 and 4 respectively – decimal places are not needed.

Then insert a further variable 'admit', which will eventually become a variable SPSS recognises as a date and which can, for example, be used is in creating derived variables such as age on admission to the current school. In the SPSS 'Variable View' window, use the Type cell for this variable to give it a DDMMYYYY date format (again see Figure 68). Use the 'Compute' procedure to create the new date. The procedure below will not work as described if the new admission date variable has not been created in advance.

In the 'Compute Variable window', follow Figure 69 and type in 'admit' as the 'Target Variable', and then scroll down the Function pane on the right hand side of the window, and left click on Date Creation. In the 'Function' and 'Special Variables' pane below this, select 'Date.Dmy'. An explanation of what this does is shown in a pane to the left. Left click on the upward arrow button immediately above that pane. DATE.DMY(,,) will be shown in the Numeric Expression pane. Between the brackets, key in newdd before the first comma, newmm immediately after the first comma and newyyyy immediately after the second comma. Click the 'OK' button and SPSS will combine the information in the three variables dd mm and yyyy in the new 'admit' date variable. Check whether what you have done has worked and, if it has, save the file.

There are further potential benefits in education research in creating three separate dd, mm and yyyy variables, and this will be the case in other fields of research. There is, for example, an association between level of attainment and season of birth. The numeric version of the mm variable can be given value labels for each month (1='January' and so on) for that type of analysis, and the next section discusses the uses of the yyyy variable in analyses of pupil mobility.

Assuming that we are working with the 2006 SC, the difference between

(a) the time from 14th October 1582 to pupil date of birth and
(b) (what should be) the slightly longer span of time up to the point where a pupil was admitted to the roll of his or her current school
(c) gives pupil age on entry to that school.

Figure 70 below shows how to calculate that time difference, and hence pupil age at the time of admission to the current school, in terms of years – that is by turning the passage of time from seconds to years (including leap years).

Figure 69. Creating a new date variable

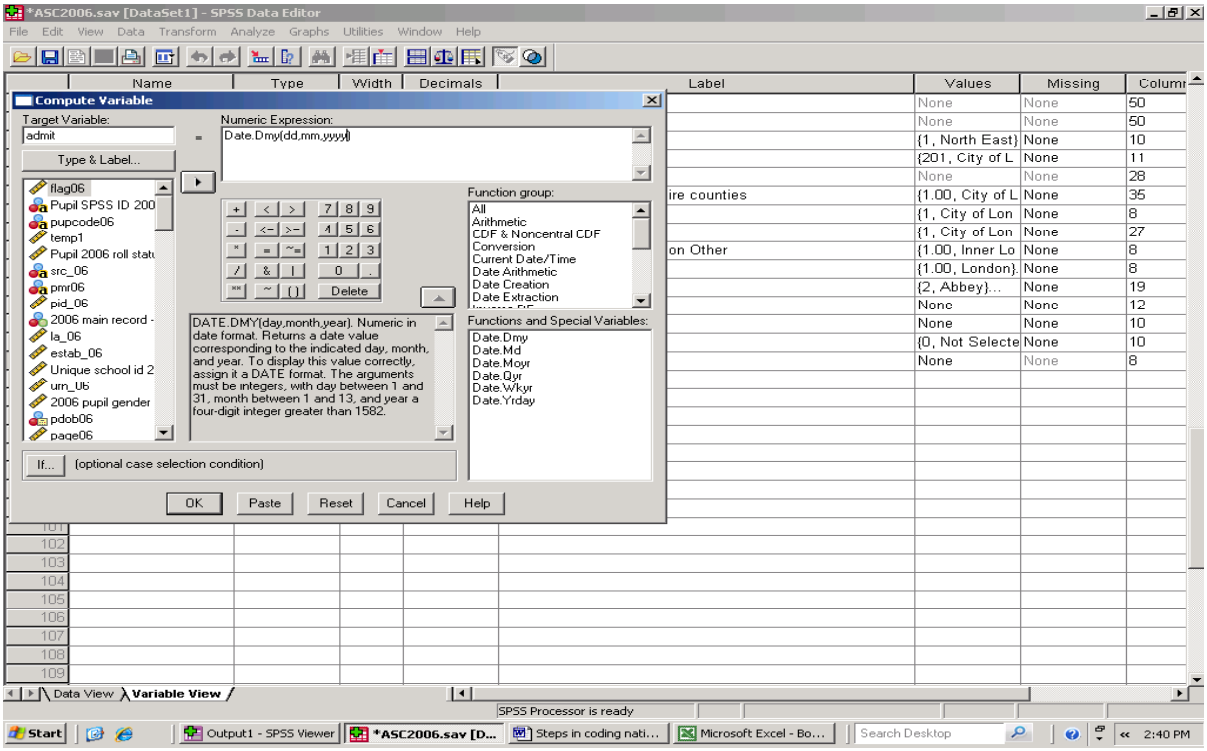
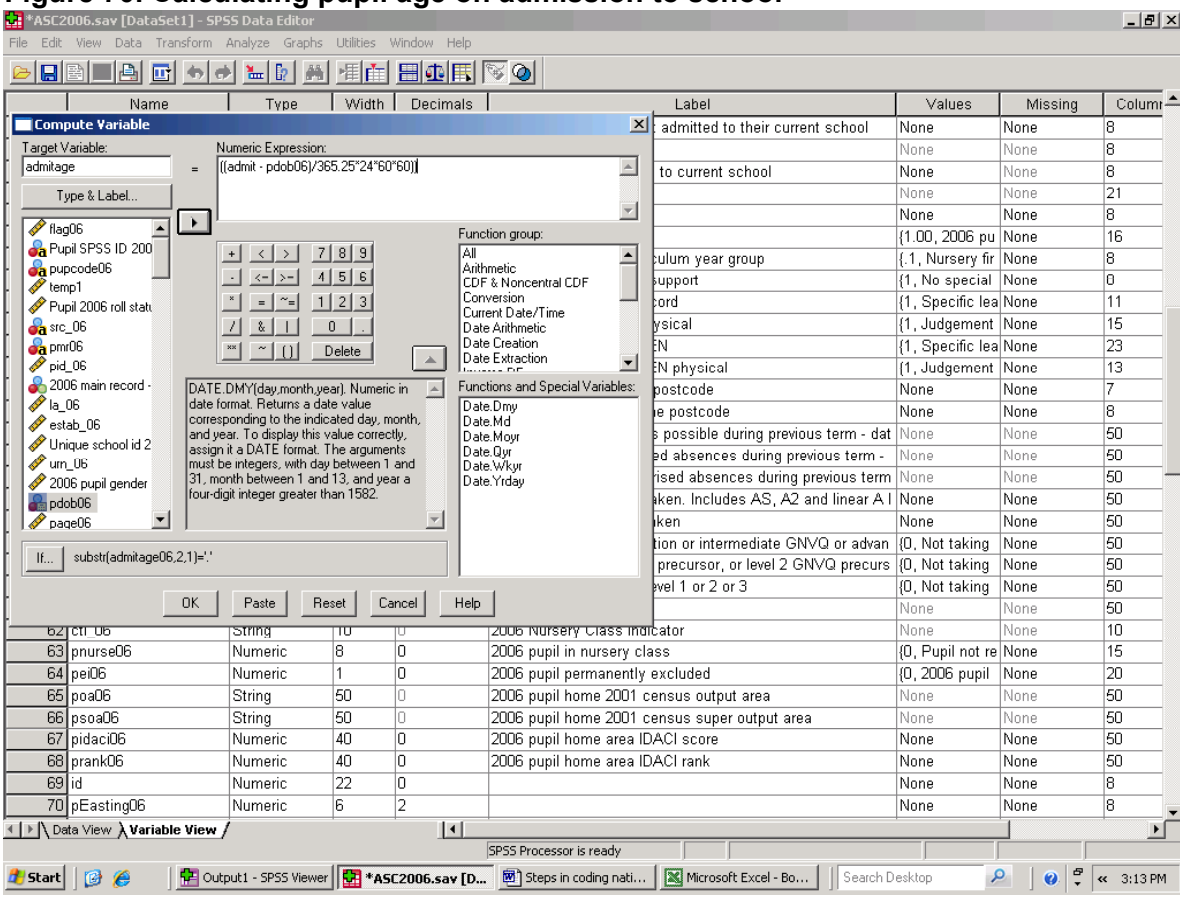


Figure 70. Calculating pupil age on admission to school



20. Inward pupil mobility. Re-basing dates, and bring together work to create new variables, calculate age, by-pass rounding numbers up, extract substrings, remove leading spaces, select particular cases and filter out others

Early sections of the Guide each focussed on a limited number of SPSS procedures, with subsequent sections gradually covering an increased range of steps that can be taken to organise data in SPSS. There will be times when only a limited number of procedures will be used in a particular exercise, but those occasions may well be few and far between. This Section focuses on the record of pupil mobility amongst six year olds in Infant Schools, to demonstrate how those procedures can be used in combination in a 'real world' setting. Mobility in other age groups is discussed towards the end of the Section. It begins though, by re-emphasising the point that knowledge of the research field in question is an essential pre-requisite to organising and analysing data in any computing software.

Studies of 'inward' pupil mobility focus on pupils who join schools at non-standard times: large numbers of inwardly mobile pupils place extra demands on a school's time, and inward mobility has been associated with below average levels of educational attainment.

A non-standard admission date would be one within the course of a school year rather than at its start. More precisely, a non-standard admission would be other than at the start of the school year and to the youngest age group the school *typically* catered for *on a full-time basis*. Given the choice of words and the emphasis in the last sentence, we might suspect that pupils admitted on a part-time basis, or who are enrolled for a part rather than the whole of a school year, can somehow be in a different position. That is indeed the case.

In England, pupils aged five at 31st August begin compulsory education in September that year, usually in a National Curriculum Year 1 primary school class. Compulsory education continues until the end of the school year which the pupil begins aged 15 at the start of the school year. Pupils in the last year of compulsory schooling will generally be in national curriculum Year 11, though it is pupil age rather than completing national curriculum year 11 that determines whether a pupil has reached the end of compulsory schooling. Partly with this in mind pupil age is given in whole years in the NPD, as it would have been at 31st August immediately before the start of the current school year, and this is so regardless of whether data are from the autumn, spring or summer surveys.

In most (but not all) London local authorities, the Reception class provides the first year of full-time education, beginning in September when a child is 4 (i.e. one year earlier than is required by law). Full-time provision for four year olds is not uncommon outside London, but admission is often staggered throughout the course of the school year as a matter of policy. In the case of schools with nursery classes, admission can also be staggered as a matter of policy.

Pupils on roll in a Nursery or Reception class may have been admitted during the course of the school year not because they are mobile in the sense of having moved home and/or changed school, but because of local policy on staggered admissions. Flagging pupils as 'mobile' when their admissions have been staggered as a matter of local policy would be an example of creating a measure which does not measure what it is intended to measure.

It is also likely to distort whatever analysis of mobility the research analyst may wish to carry out, and that distortion will not only apply to pupils currently on roll in nursery or reception classes. Pupils will retain the same admissions date until they leave their current school, which means that in schools with nursery or reception age groups, at least some pupils in all age groups may have a non-standard admissions dates because of staggered admissions policies applied to pupils admitted at a young age.

Figure 71 shows the age at the start of the 2007/08 school year of pupils attending English maintained schools. The Table also shows the age of those pupils as at 31st August prior to the start of the school year when they were admitted to their current school. More than 3 million of the total 7.5 million pupils on roll in 2008 had been admitted to their current school at age 4 or less. We cannot know from the data in the NPD which of the pupils who were admitted below the age of compulsory schooling and during the course of the school year were mobile and which were admitted as a matter of local staggered admissions policies. However, what we can do is to identify those pupils who were on roll at the start of the first year of compulsory education, that is when they would have been aged 5, and when staggered admissions policies would not have been in force.

Pupil age as it would have been at the start of the school year when they were admitted to the current school is not included in the data extracts released by DCSF. That age is calculated by subtracting data of birth from 31st August immediately before the school year in which the

Figure 71. Age of pupils on roll in 2008 and age when admitted to current school

	Pupil age at start of the 2007/08 school year										
	0	1	2	3	4	5	6	7	8	9	10
*Age when admitted to current school**											
Missing data	8	6	56	118	133	167	221	159	121	49	53
0						1	1	6	1		
1	56	301	776	689	258	281	368	602	395	360	411
2			44,094	63,761	41,339	36,621	33,125	25,879	23,103	18,380	15,156
3				212,866	153,611	136,540	126,503	96,027	91,537	82,277	70,540
4					352,068	323,039	297,792	225,954	215,807	195,015	199,993
5						36,273	46,084	30,795	26,350	23,975	23,028
6							34,091	37,950	30,089	24,559	24,530
7								130,073	136,335	123,557	117,899
8									39,101	47,374	41,077
9										53,440	61,998
10											29,583
Total	64	307	44,926	277,434	547,409	532,922	538,185	547,445	562,839	568,986	584,268

	2008 ASC pupil age at start of school year (continued)											
	11	12	13	14	15	16	17	18	19	20	21+	Total
*Age when Admitted to current school**												
Missing data	2	7	6	5	8	4	2	2				1,131
0												10
1	52	50	77	65	79	51	44	34				4,949
2	230	239	215	214	221	169	143	129	1			303,019
3	432	287	304	323	294	216	202	155	4			972,118
4	1,374	535	557	490	457	278	212	168	3	4	1	1,813,747
5	562	311	309	303	278	139	107	107	4		1	188,626
6	361	214	257	256	286	103	76	66	5	1	2	152,846
7	521	412	378	360	375	153	106	75	3	1	4	510,252
8	4,050	593	347	327	423	127	110	53	5	4	1	133,592
9	21,615	19,901	422	430	464	143	92	87	4	3	1	158,600
10	9,268	7,057	4,605	4,786	6,379	1,001	514	143	5	4	3	63,348
11	528,133	514,256	496,251	468,833	468,460	117,734	94,244	5,372	99	8	7	2,693,397
12		31,147	37,700	33,578	33,456	7,301	4,765	612	28	3	1	148,591
13			46,107	51,427	46,253	14,748	11,564	781	16	5	1	170,902
14				26,347	30,715	7,329	4,620	550	25	3	2	69,591
15					11,837	3,992	1,601	421	52	6	2	17,911
16						55,722	35,820	2,823	120	16	4	94,505
17							8,392	3,460	225	24	4	12,105
18								2,913	365	34	4	3,316
19									235	52	20	307
20										48	7	55
21											19	19
22											6	6
24											1	1
28											1	1
53											1	1
Total	566,600	575,009	587,535	587,744	599,985	209,210	162,614	17,951	1,199	216	93	7,512,946

Source: January 2008 English Pupil Dataset

pupil was admitted to the current school. This is a more complex exercise than the one shown in Section 19, but the steps involved are all logical – or in plain English, each step makes sense and can be understood.

Continuing to work with January 2008 data, we will assume that the format of the date of birth and school admission date records have the same format as the dates referred to in Section 19. We will also assume that the substring function has been used to create dd, mm and yyyy variables for date of birth and admission dates, and that these have been combined to create what SPSS will recognise as date variables.

The next step is to insert three dummy numeric variables, admitdd08a, admitmmo8a, and admityyyyy08a, near the source variables within the dataset. These have the same character as the dd, mm and yyyy variables described in Section 19. Keeping the source and dummy variable close together enables visual checks on data to be carried out as work progresses.

The objective is to

- create a dummy admission date
- showing admission to be at the beginning of the school year in which the pupil reached the age of 5
- in those cases where children had been admitted at to their current school when aged less than 5.

In the first instance, this requires calculating pupil age in whole years at the start of the school year in which he or she was admitted to the current school, and on the basis of which we can identify pupils aged below the age of compulsory schooling. We will call the variable showing the

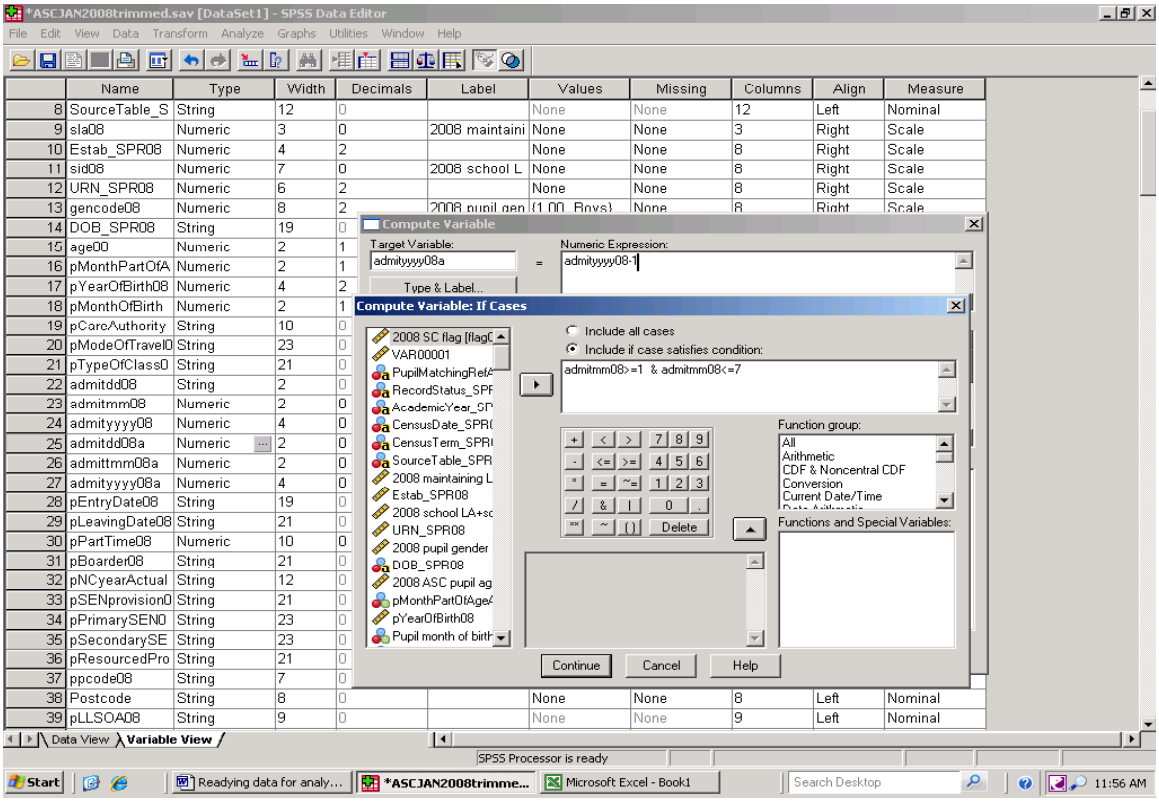
calendar year in which the pupil was admitted "admityyyy08". However, the admission year does not necessarily show the year at 31st August prior to start of the school year school in which a pupil was admitted to his or her current school.

The school year is split between two calendar years, running from September to July, and age is calculated as it would have been in whole years at 31st August immediately before the start of the school year. Pupils admitted between January and July will have an admission year one year 'after' the one needed. In those cases admission year will need to be reduced by one. If source data are needed for other purposes, create a copy of the admission variable (if you are working as part of a team, always check before modifying or deleting source variables)

Compute admityyyy08a = admittyy08

Then
Compute admityyy08a=admityyy08 minus 1 if admitmm08 is >= 1 and admitmm08 is <=7.

Figure 72. Calculating the year in which a child's first school year began

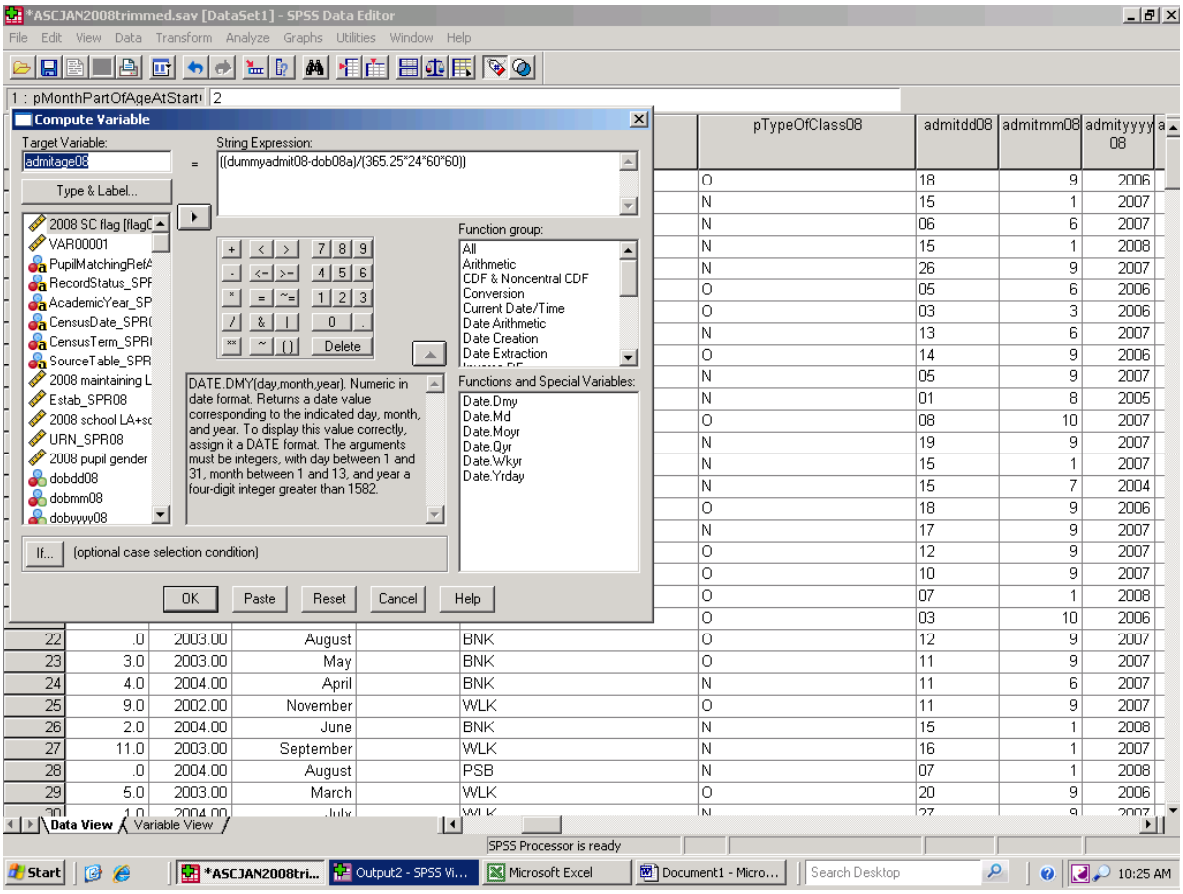


The second of the two steps is shown in Figure 72. In plain English, all pupils have been given a year of admission which is the calendar year of the first day of the autumn term of the school year in which a child was admitted. You now have one of the three variables needed to create a dummy admission date at the start of the appropriate school year. To create the other two elements of the dummy admission date, create one dummy numeric variable for "31st" and another for "August".

Compute admitdd08a=31
and
Compute admitmm08a=8

Combine the three dummy variables to create an admission date as at 31st August at the start of the school year in which pupils were admitted. Then calculate pupil age as it would have been at the start of the school year admitted, by subtracting date of birth from dummy admission date (see Figure 73 below).

Figure 73. Calculating age as it would have been at the start of the school year in which pupils were first admitted to their current school

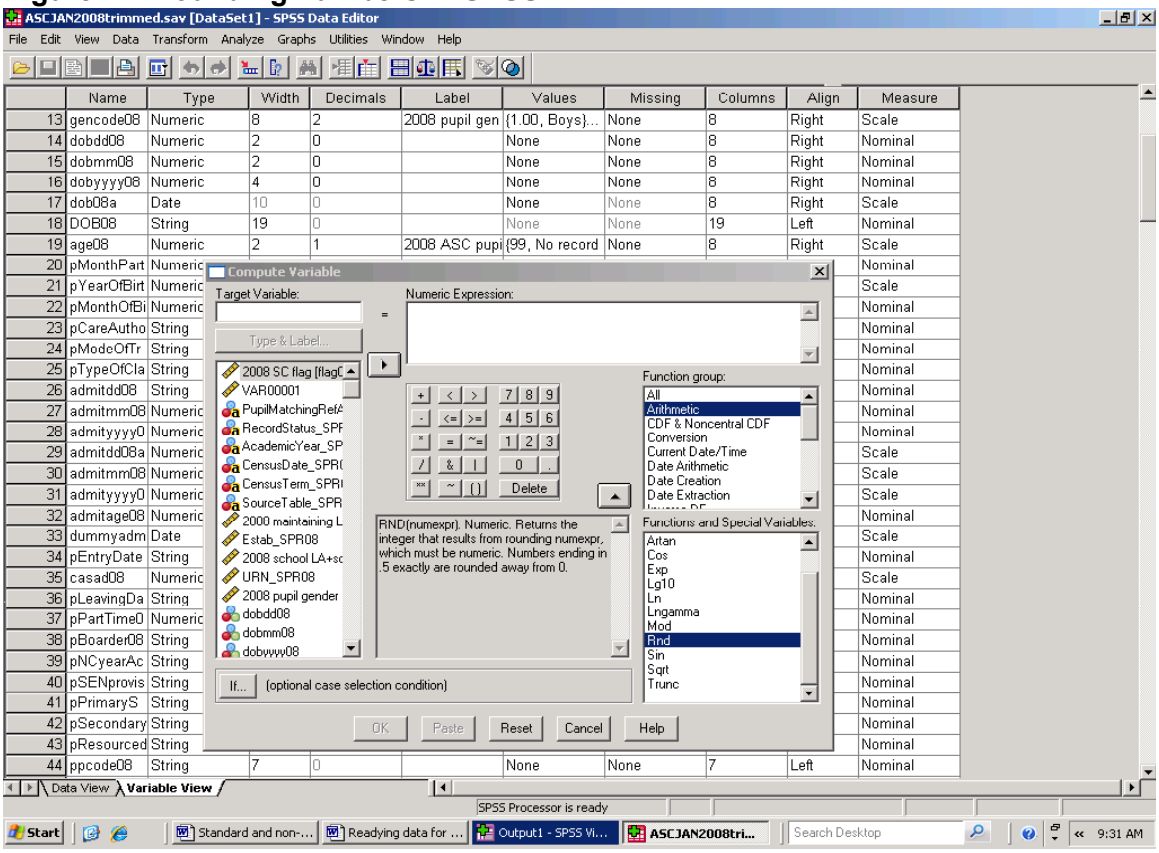


Pupil age will be shown to as many decimal places as you have set, and a reasonable question is what bearing the numbers on the right hand side of the decimal point have for those on the left. Do we round up, down or what? SPSS has a rounding function in its 'Compute function group', as shown in Figure 66 – and it is *not* used on this occasion. More specifically, we do not want SPSS to round the dummy age variable up.

The explanation is that compulsory schooling in England begins immediately after the 31st August when child is aged five, and for the purposes of educational administration age is recorded in unrounded whole years. Twelve months before a pupil reaches that age 5 at 31st of August, he or she would be aged 4, and would not be of compulsory school age. This in turn means that pupils aged 5 at 31st August could have just reached their birthday at that point, or could have had their fifth birthday eleven months before hand. Pupils can differ in age by almost, but not quite, twelve months and still be in the same age group when viewed in terms of education administration. Put another way, in age measured to one or more decimal points, figures to the right of the decimal point have no bearing on the numbers to the left of that point. Pupil age in whole years needs to be extracted unchanged from the information SPSS has just calculated for us. (The potential disparity of nearly a year in the actual age of pupils in the same age group goes some way to explain why summer born children are so often 'behind' autumn and winter born

children in cognitive development in the early years of primary schooling: the latter have had almost one full year more in which to develop. What is less clear is why disparities in the attainment of summer and autumn born children persist up to the end of compulsory schooling and beyond.)

Figure 74. Rounding numbers in SPSS



It would be prudent to run a Frequency Table on the age variable as SPSS has calculated it before going further. There will be some, but not many ages at the start of the school year admitted which cannot be correct, and Table 71 confirms that in 2008 there were slightly more than 1,000 pupils (in a dataset of 7.5 million pupil records) where the record of date of birth either cannot be right or is missing. As far as work with a full national extract from the NPD is concerned, there is nothing an individual research analyst can do to change those 1,000 records in the absolute certainty of being correct. Section 7 does, however, offer one way of making a fair guess at what missing records of age might actually be.

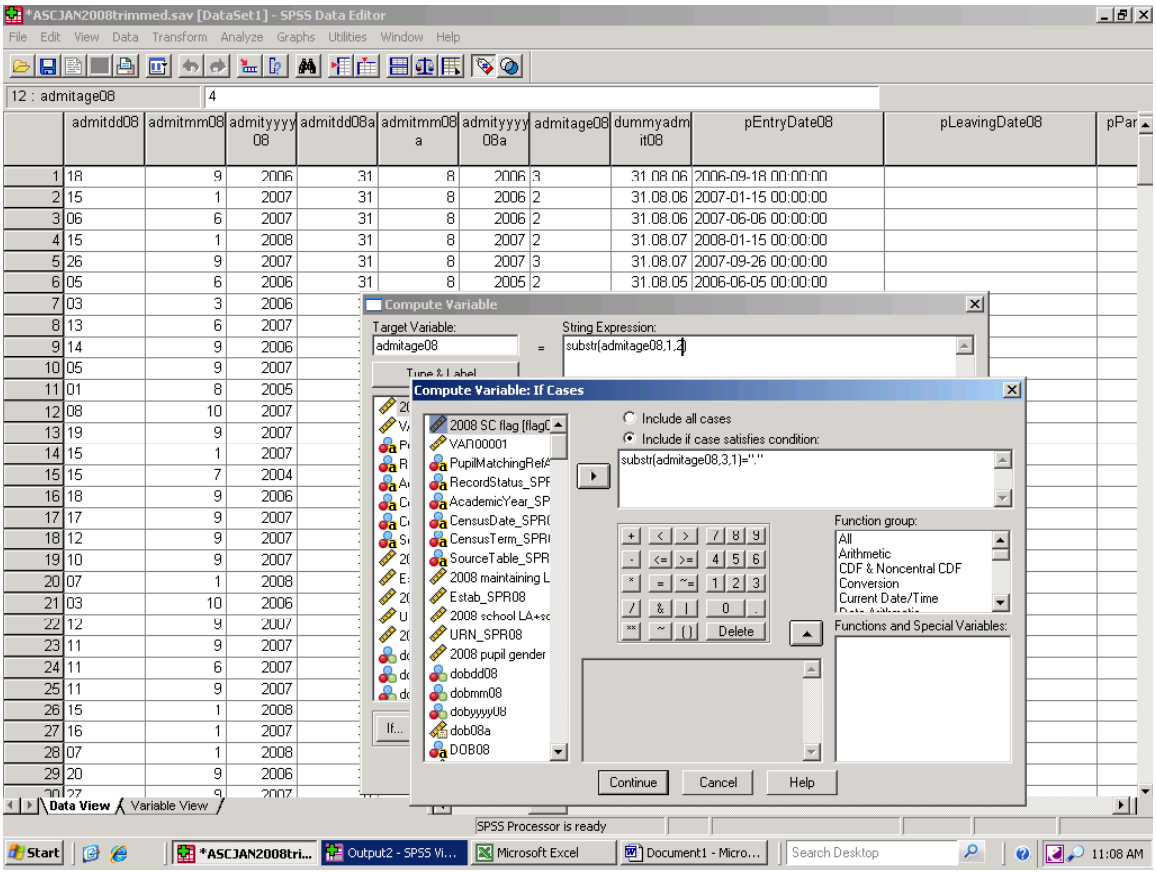
Accepting that there is a margin of error in the data, we need to extract only that information on age to the left of the decimal point. This will involve a minimum of one number (for example for pupils aged seven) and a maximum of two numbers (for example for pupils aged 11).

If admission age is changed to a string variable, the Ltrim function can be used to remove any leading spaces, and there will then be only two places within the age string where the decimal point will lay. For pupils aged 11 and over, the decimal point will be the third character in the string and for pupils aged nine and under the decimal point will be the second character in the string. The substring function with a conditional 'If' statement, as illustrated in Figure 75, can now be used to create a record of pupil age at the start of the school year in whole numbers and without and rounding up. A modified version of the instructions in Figure 75 to take only the first character of the string if the second character is a decimal point will give the age in whole years of pupils aged 9 and less. (If this is needed, Section 10 provides more detailed information on working with strings). The 'admitage08' Variable can be converted to a numeric form in SPSS's Variable View by selecting the 'Type' cell next to the left of the variable name and then by selecting numeric. You now have a record of the in whole years of pupil age when they were first admitted to their current school.

Have established the new age variable, the next steps will reflect the research question in hand. Another dummy admission date might be created, set at 31st August and set one year than that already shown for pupils admitted when aged four, and two years later for pupils admitted when aged 3. (That is, in both cases the dummy

admission year is set at the start of compulsory schooling when the pupil would have been aged 5). Alternatively, a binary 'yes/'no' variable might be created showing whether a pupil had a non-standard admission month, and in which pupils aged on roll (in this case in 2008) who had been admitted when below the age of compulsory schooling were not flagged as having a non-standard admission date.

Figure 75. Extracting the record of whole years of age without rounding up



The example shown here focuses on avoiding inflating the number of pupils aged 6 at the start of the year because they were admitted within, rather than at the beginning of the school year as the result of local. This involves creating several dummy variables, which will have been given a name in the appropriate 'Label' cell as work progresses to ensure their meaning is not lost. Can the same procedures be used to avoid inflating pupils with a record of casual admissions in other age group? They can be used in that way as long as care is taken to ensure that the point made in the third paragraph of this Section is understood "a non-standard admission would be other than at the start of the school year and to the youngest age group the school *typically* catered for *on a full-time basis*."

In terms of the compulsory school age range, the youngest age group typically catered for by an English Infant School, or an English JMI (Junior and Mixed Infant) school, and in some special schools, will typically be aged five at the start of the school year. However, the youngest age group typically catered for in English Junior schools will, typically, be aged seven at the start of the school year, and the youngest age group in a secondary school will, typically, be aged 11 at the start of the school year. Figure 76 summarises the age range schools catered for in 2008, and the situation is more complex than the '5, 7, 11' model might suggest. There are other school types which the research analyst dealing with casual admissions other than to nursery and reception classes will need to be aware of, and that information is not given by SPSS (or any other statistical package). Knowledge of procedures that are useful in organising data is not the same as, or stands as a substitute for, knowledge of what the data mean.

Figure 76. School statutory low age by statutory high age 2008, by pupil numbers

	High age															
	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
low age																
2		60	84	67			93	6,426	374	60	140		2,065	448	456	16,864
3	417	37,080		187,140	8,446	48,510	1,033	1,637,481	4,965	233	79		4,564		3,617	10,932
4				26,022	104	2,770	7,063	1,110,156	6,004	89	97		4,461	466	1,907	3,073
5				126,518	11,498	61,150	6,845	423,442	2,241		104		5,368	91	123	2,115
6								44		52			578		93	150
7							48	406,251	34		37	96	3,761		1,385	470
8								1,516	15,361		48		586	65	1,682	109
9									1,420	89,917			542	26	40	86
10										4,039	5,577		443	62	52	1,773
11											17,760		1,040,806	3,727	1,896,217	94,679
12													8,249		18,071	136
13													66	163	95,185	2,017
14													72		16,817	5,106
16															1,231	2,518
Total	417	37,140	84	339,747	20,048	112,430	15,082	3,585,316	30,399	94,390	23,842	96	1,071,561	5,048	2,036,876	140,028

Source: January 2008 English Pupil Dataset

21. Calculating straight line distance between two points using Northings and Easting

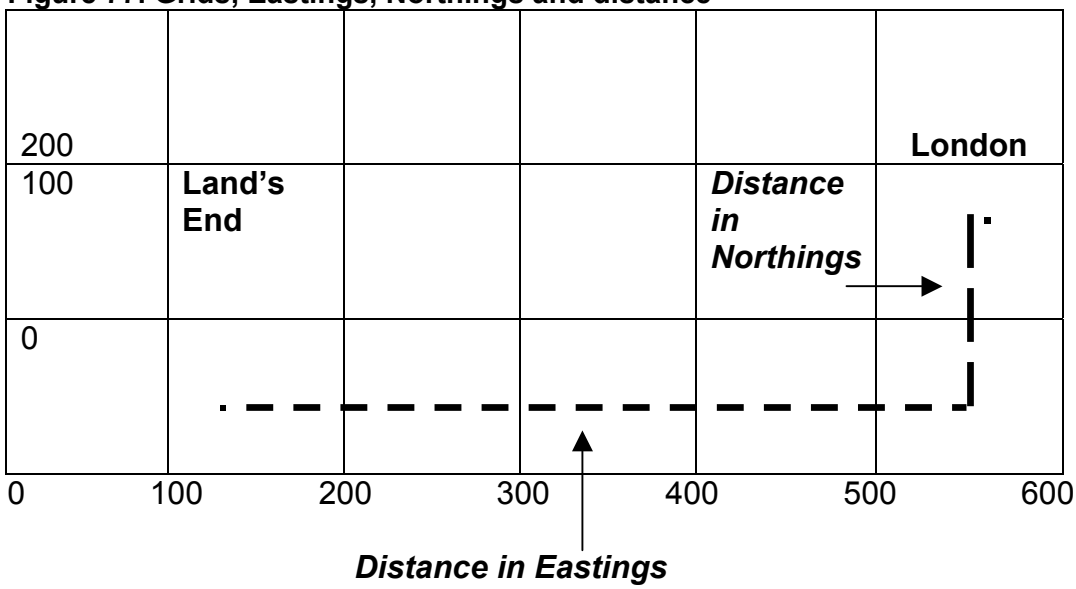
Questions about distance can arise in a variety of contexts. How far do people travel to work? How far do people move when they move home? What is the distance between where people live and their nearest medical practice or hospital? What is the distance between where people live and the nearest open space, railway station, port or airport?

Questions of distance also have their place in education research. Do pupils who attend schools in local authority areas other than the one in which they live simply attend a nearby school (i.e. travel short distances) or do they travel longer distances than other children living in the same local authority area? Do pupils attending popular schools tend to live close to those schools, or do they live at some distance from them? How far do children living near popular schools actually travel to get to school?

Clearly there will be cases where the simple straight line distance as measured between two points on a map is not necessarily very useful, other than by way of providing a comparison with the actual distance travelled to reach hospitals/ shops/ schools and so on. A cursory glance at the road and rail maps of the more mountainous regions of Britain will confirm that this is so. For example, a journey wholly by rail from Mallaig (on the north west Scottish coast and opposite Skye) to the Kyle of Lochalsh (a short distance to the north east but still next to Skye) would involve travelling via Perth (on Scotland's east coast) and Inverness (on Scotland's north east coast). If you need complex measures of distance, you are advised to seek help from a Geographic Information Systems (GIS) specialist. On the other hand, if measuring straight line distances serves a useful purpose, these can be calculated as long as the dataset being used contains the necessary national grid references.

British national grid references begin at zero at a point to the west of the Scilly Isles, south west of Land's End. Distance is then measured to the east (in 'Eastings') and to the north (in 'Northings'). A six-digit Easting measures distance in metres to the east of that point near the Scilly Isles. A five digit Easting measures distances in units of 10 metres to the east of zero, and a four digit Easting measures distances in units of 100 metres from zero. Likewise, a six-digit Northing measures distance in metres to the north of zero, and so on.

Figure 77. Grids, Eastings, Northings and distance



The pupil datasets referred to here contain grid references for both pupil home postcode and for the postcode of the school attended.

Assume that a pupil living at Land's End in Cornwall attends a school, perhaps a special school, in London, which is further east and further north than Land's End. The distance in Eastings between the child's home and school forms one side of a triangle, and the distances in Northings forms the second side of a triangle. The straight-line distance between the child's home and the school is the hypotenuse of that triangle. The length of the hypotenuse is the straight line distance between the child's home and the school attended, and that distance can be calculated using Pythagoras' theorem that the square on the hypotenuse of a triangle is equal to the sum of the squares of the other two sides of the same triangle.

On that basis, the square of the hypotenuse is the square of the distance in Eastings plus the square of the distance in Northings, and the straight-line distance between the child's home and the school attended is simply the square root of the square of the hypotenuse.

Broken down into the component stages the calculation is

((school Easting minus pupil Easting) multiplied by (school Easting minus pupil Easting))

plus

((school Northing minus pupil Northing) multiplied by (school Northing minus pupil Northing))

equals the square of the straight line distance between the pupil's home and the school attended

and the square root of that value is the straight line distance between the two.

The 2007 pupil dataset is used here to give a worked example of that calculation. The variable containing pupil home Easting in 2007 is peasting07, and the variable containing the Easting of the school attended is seasting07. We do not know whether a pupil's home is to the east of the school on the national grid or to the west, and simply subtracting peasting07 from seasting07 will produce at least some negative eastings. This does not matter in this instance. What we actually want is the *square* of the distance in Eastings. That is given by multiplying

- (*peasting07* minus *seasting07*) by (*peasting07* minus *seasting07*).

That multiplication provides the square of the distance in Eastings, and in doing so automatically converts negative numbers to positive numbers.

Similarly, multiplying

- (pnorthing07 minus snorthing07) by (pnorthing07 minus snorthing07)

gives the square of the distance between pupil home and school in 2008 in Northings, and Figure 78 shows that calculation. The square of the straight-line distance between home and school is, as noted above, simply the sum of the squared home-school distance in Eastings (hseasting07) and the squared home-school distance in Northings (hsnorthing07).

The square root of (hseasting07 + hsnorthing07) is the straight-line distance between home and school in metres. Figure 79 shows the calculation, and illustrates a further point. The 'Compute Variable' window includes a pane headed 'Function group'. The function group 'Arithmetic' has been highlighted, and below that, the 'Function and Special Variables' list the arithmetic function which can be selected. In this case 'Sqr' is highlighted, and the 'greyed' pane to the left explains, that the function calculates the square root. The black upward pointing arrow can be used to transfer the function name to the 'Numeric Expression' box, complete with following brackets within which variable names and numeric expressions can be keyed. (You could, of course, type in the full expression into the 'Numeric Expression' pane and skip the 'point and click' stages). Whichever route is taken in this instance, a key point is that there are a number of functions in SPSS which can be accessed through the 'Function group' and 'Functions and Special Variables' panes, and that at least some explanation of what the functions mean is given in the greyed pane shown in Figure 79.

Dividing home school distance in metres by 1,000 gives the distance in kilometres, and dividing by 1,609.344 gives the distance in miles (there are 1,609.344 metres in a mile). The unit of measurement chosen will reflect both particular research objectives and the audience being addressed. Some people find it difficult to think of distance in terms of kilometres while others struggle with distances measured in miles.

Figure 80 shows an early calculation of average home-school distance for pupils attending mainstream schools maintained by their 'home' local authority and average home-school distance for pupils attending mainstream schools maintained by a local authority other than the on in which they live in 2008. Pupils attending schools maintained by another authority might be expected to travel further than pupils attending a school maintained by the 'home' local authority. Where Figure 80 indicates that this is not so, as is the case in Milton Keynes, it may well be because of inaccuracies in the raw data or in the way in which distance has been calculated. It provides a rather neat example of the way in which SPSS Tables can be used to check data quality and help pinpoint likely errors.

Figure 78. Calculating straight line distance in Northings

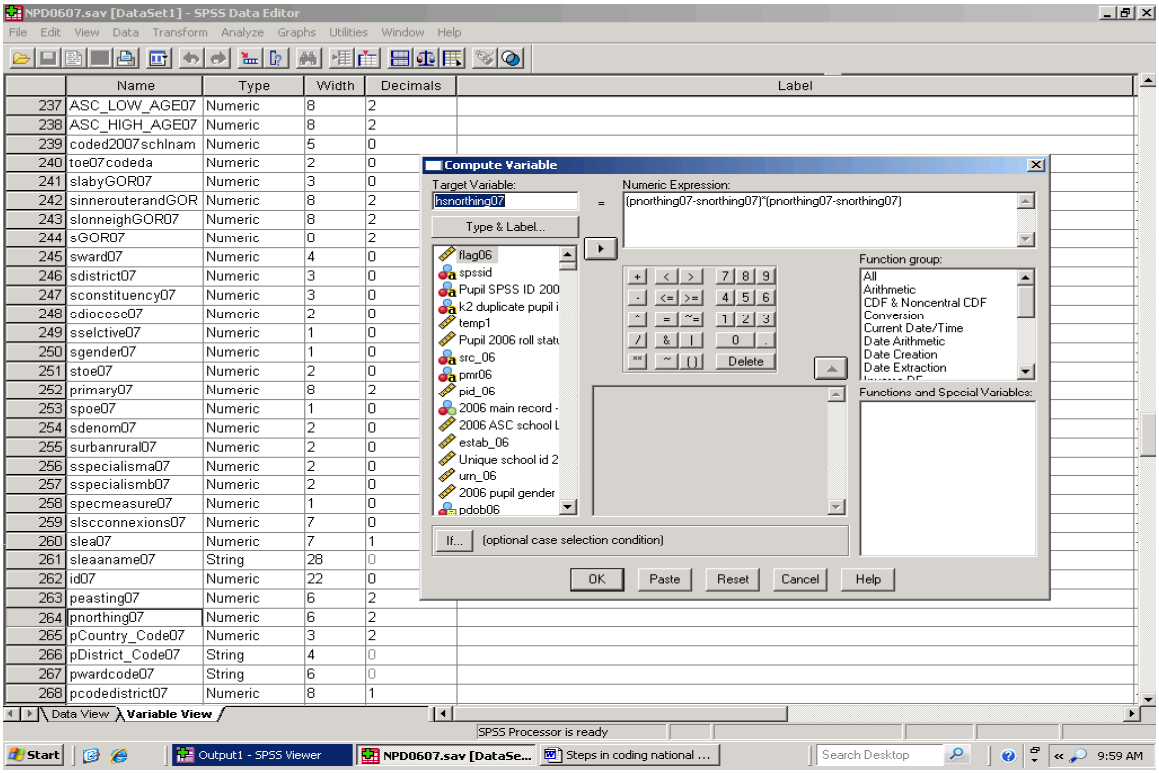


Figure 79. Calculating the square root of the hypotenuse

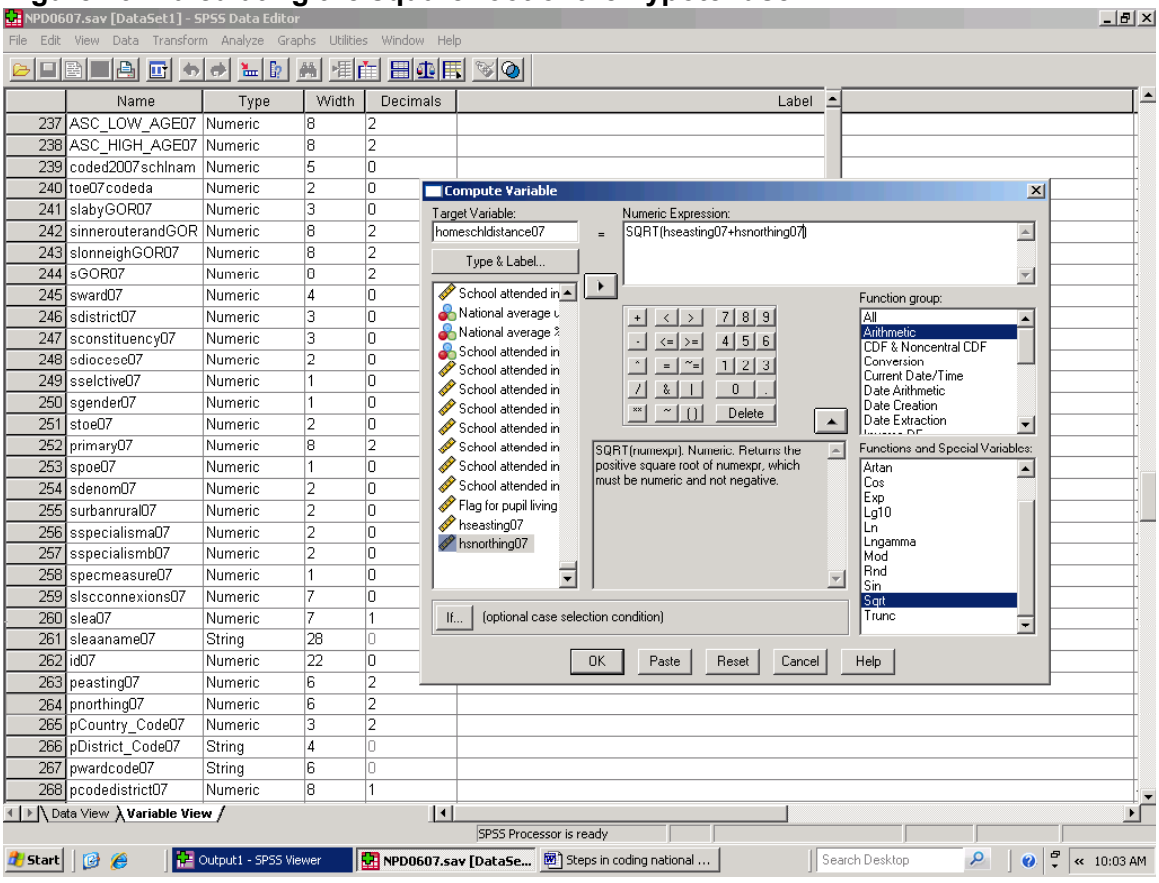


Figure 80. Unedited average home-school distance as the crow flies. Mainstream schools, excluding overseas schools. Pupils attending schools maintained by the home LA and pupils attending schools maintained by another LA

	Home school distance in miles in 2008			Home school distance in miles in 2008	
	School attended is not maintained by pupil's home LA	School attended is maintained by pupil's home LA		School attended is not maintained by pupil's home LA	School attended is maintained by pupil's home LA
Pupil home LA by inner/outer London and GOR, 2008					
City of London	49.4	0.3	Buckinghamshire	14.5	1.8
Camden	1.9	0.6	East Sussex	8.4	1.4
Hackney	2.6	0.5	Hampshire	4.9	1.2
Hammersmith and Fulham	2.8	0.6	Isle of Wight	38.2	1.7
Haringey	4.1	0.7	Kent	5.4	2.5
Islington	2.3	0.6	Medway	6.1	1.1
Kensington and Chelsea	2.7	0.6	**Milton Keynes**	**7.1**	**20.2**
Lambeth	2.6	0.7	Oxfordshire	19.0	2.1
Lewisham	3.5	0.7	Portsmouth	3.5	0.6
Newham	5.3	0.5	Reading	2.2	0.7
Southwark	**2.7**	**9.0**	Slough	3.5	0.8
Tower Hamlets	6.1	0.5	Southampton	4.3	0.7
Wandsworth	2.8	0.7	Surrey	4.3	1.3
Westminster	2.4	0.6	West Berkshire	4.7	1.3
Barking and Dagenham	**3.3**	**12.5**	West Sussex	5.9	3.7
Barnet	3.3	2.9	Windsor and Maidenhead	4.1	1.0
Bexley	7.2	0.9	Wokingham	3.6	1.0
Brent	2.9	0.9	Bath and North East Somerset	5.3	1.2
Bromley	4.0	1.1	Bournemouth	2.8	1.0
Croydon	2.9	1.0	City of Bristol	3.0	1.9
Ealing	3.1	0.8	Cornwall	10.1	1.9
Enfield	3.4	0.9	Devon	6.4	2.0
Greenwich	2.9	0.8	Dorset	4.5	2.3
Harrow	3.3	0.8	Gloucestershire	7.6	2.3
Havering	6.7	0.9	Isles of Scilly	221.2	1.0
Hillingdon	**4.3**	**7.1**	North Somerset	7.4	1.1
Hounslow	3.0	0.8	Plymouth	6.2	0.9
Kingston-upon-Thames	2.6	0.8	Poole	2.5	1.0
Merton	2.4	0.8	Somerset	5.8	1.6
Redbridge	3.9	0.8	South Gloucestershire	4.6	0.9
Richmond-upon-Thames	2.9	0.7	Swindon	4.7	2.6
Sutton	2.6	0.8	Torbay	5.3	0.9
Waltham Forest	3.2	0.6	Wiltshire	7.5	1.5
Bedfordshire	6.2	1.2	**City of Derby**	**7.4**	**14.1**
Cambridgeshire	14.2	1.4	Derbyshire	39.8	2.4
Essex	6.0	2.4	City of Leicester	2.6	0.7
Hertfordshire	5.5	2.3	Leicestershire	6.6	3.5
Luton	5.6	0.7	Lincolnshire	6.1	2.1
Norfolk	6.4	1.7	**Northamptonshire**	**8.4**	**12.0**
Peterborough	**6.9**	**7.9**	City of Nottingham	2.5	0.7
Southend-on-Sea	3.6	0.9	Nottinghamshire	5.3	1.4
Suffolk	9.5	2.1	Rutland	9.5	1.7
Thurrock	7.8	0.8	Birmingham	2.5	0.9
Bracknell Forest	4.0	0.8	Coventry	4.1	0.8
Brighton and Hove	7.3	0.9	Dudley	2.7	0.7

Source: 2008 English National Pupil Dataset

Figure 81. Unedited home-school distance as the crow flies. Mainstream schools, excluding overseas schools. Pupils attending schools maintained by the home LA and pupils attending schools maintained by another LA - continued

	Home school distance in miles in 2008			Home school distance in miles in 2008	
	School attended is not maintained by pupil's home LA	School attended is maintained by pupil's home LA		School attended is not maintained by pupil's home LA	School attended is maintained by pupil's home LA
Pupil home LA by inner/outer London and GOR, 2008					
Herefordshire	7.5	2.0	Halton	4.0	0.7
Sandwell	2.4	0.7	Knowsley	2.5	0.7
Shropshire	9.2	1.6	Lancashire	5.1	1.2
Solihull	4.0	0.8	Liverpool	3.1	1.0
Staffordshire	5.0	1.1	Manchester	2.2	0.8
Stoke-on-Trent	2.5	0.7	Oldham	2.5	0.7
Telford and Wrekin	**5.0**	**14.1**	Rochdale	3.7	0.8
Walsall	2.8	0.8	Salford	2.6	0.7
Warwickshire	6.1	1.3	Sefton	3.5	3.3
Wolverhampton	3.6	0.8	St Helens	2.5	0.9
Worcestershire	4.9	3.8	Stockport	3.5	0.8
Barnsley	4.0	0.9	Tameside	3.4	0.8
Bradford	3.2	1.9	Trafford	4.8	0.9
Calderdale	5.5	1.1	Warrington	5.6	0.9
Doncaster	7.1	1.0	Wigan	2.8	5.3
East Riding of Yorkshire	6.7	1.4	Wirral	4.4	1.1
City of Kingston Upon Hull	2.4	0.7	**Darlington**	**9.5**	**10.3**
Kirklees	3.0	0.9	Durham	6.5	1.3
Leeds	7.2	5.9	Gateshead	3.7	0.9
North East Lincolnshire	10.0	0.8	Hartlepool	15.0	0.8
North Lincolnshire	8.2	1.2	Middlesbrough	5.8	0.7
North Yorkshire	7.4	1.8	Newcastle-upon-Tyne	3.7	0.9
Rotherham	4.7	3.4	North Tyneside	3.2	0.8
Sheffield	3.0	0.9	Northumberland	7.6	1.7
Wakefield	4.5	0.9	Redcar and Cleveland	5.5	0.9
City of York	9.2	0.9	South Tyneside	4.2	0.8
Blackburn with Darwen	4.9	0.8	**Stockton-on-Tees**	**5.2**	**25.3**
Blackpool	3.5	0.8	Sunderland	4.2	0.8
Bolton	4.2	0.9	Other UK country e.g. Scotland	13.2	.
Bury	2.5	0.9	No postcode match	.	.
Cheshire	6.5	1.2	Total	4.8	2.1
Cumbria	23.0	1.6			

Source: 2008 English National Pupil Dataset

22. Aggregating data. Grouping information for different cases

In SPSS English, each row is a case, and the case is the unit of analysis. In the pupil datasets, the pupil record forms the case, and therefore the unit of analysis. However, in the public examination subject entry files, each row, that is each case, is an individual examination entry.

This point was introduced in Section 3 (page 9). If a pupil takes 10 examination subjects, there will be 10 rows of information for that individual. The dataset also contains a unique code for the pupil taking those examinations and we can, for example, determine how many subjects each candidate sat by using the Aggregate procedure.

In this worked example, Aggregate procedures will be used to create a small dataset with the total number of examination entries for each candidate, the average point score for all examinations entered by each candidate, and the total point score gained by each candidate. Examination point score reflect the grade achieved and the type of examination taken. The higher the grade, the higher the point score, and more points can be scored in a full subject entry than in a part subject entry.

To access the 'Aggregate' procedure, select 'Data' from the main SPSS menu, followed by 'Aggregate'. This produces the 'Aggregate Data' dialogue box illustrated in Figure 82, which has the typical SPSS variable list on the left, and two transfer arrows immediately to the right of the list. Allocating a variable to the 'Break Variable(s)' section of the dialogue box means that data will be grouped together for that variable. In this example, information will be summarised for each (anonymised) candidate. Three variables have been transferred to the 'Summaries of Variable(s):' section. The default position, shown in Figure 82, is for SPSS to calculate the mean for each of the three variables.

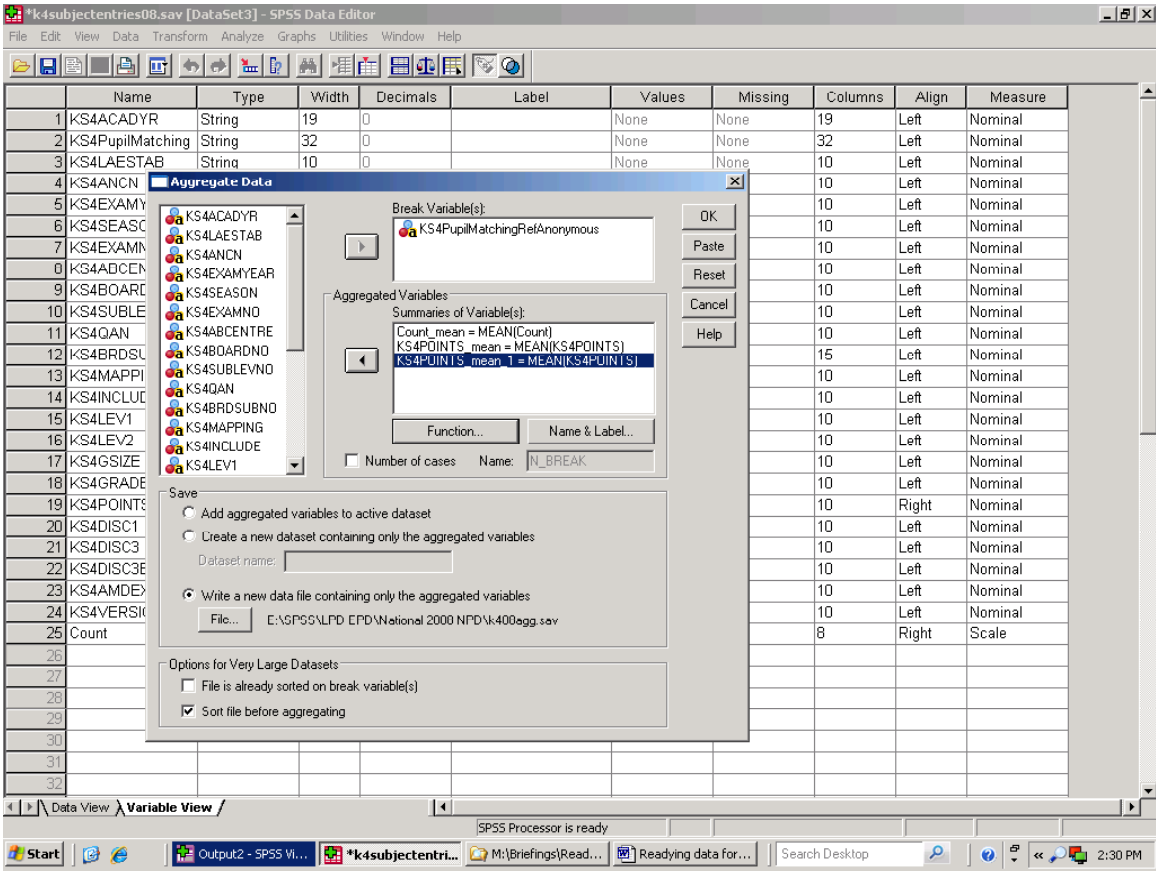


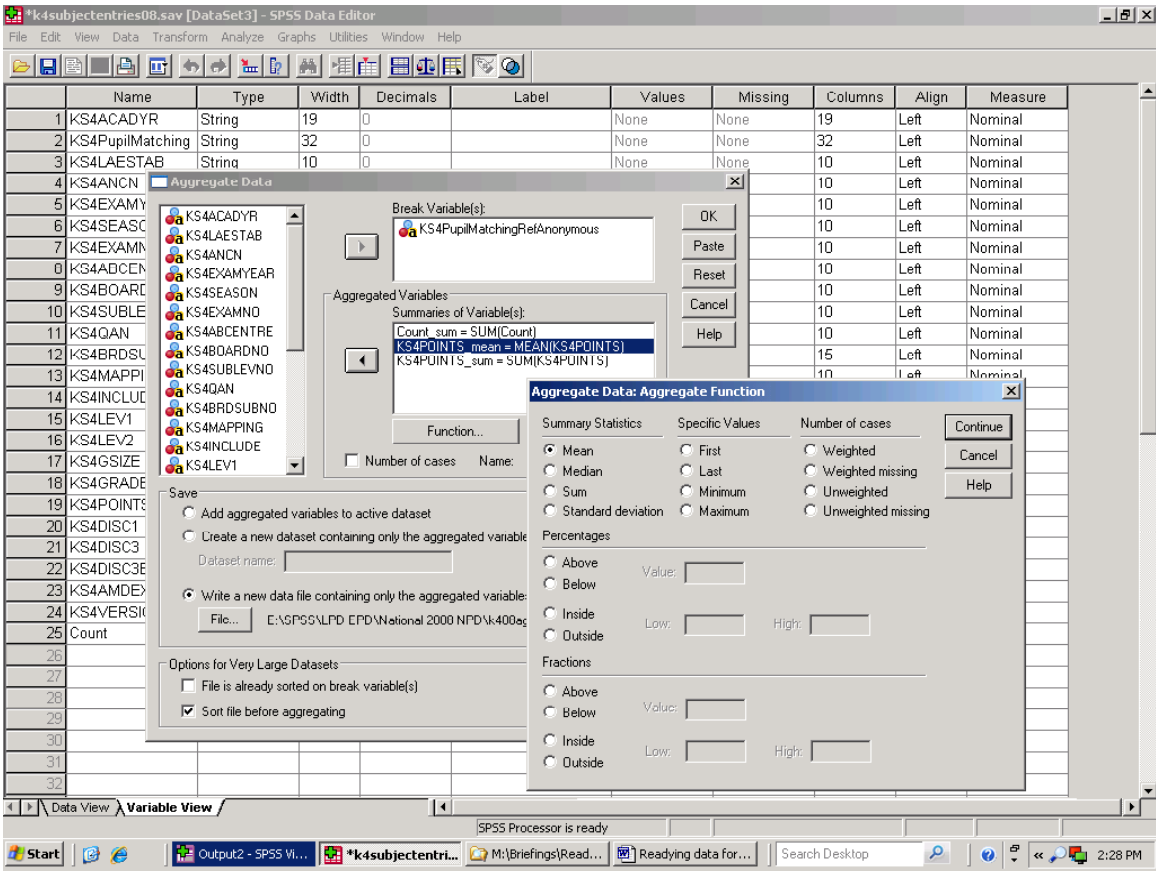
Figure 82. Aggregate Data: Aggregate Function dialogue box

Left clicking on a variable in the 'Summaries of Variable(s):' section, followed by selecting the 'Function' button immediately below the 'Summaries of Variable(s):' section, enables the user to change the way in which data are summarised. The options available are shown in Figure 83. The first variable will be summed (counted), and will provide each candidate's total number of examination entries per candidate. The variable has been added to the subject entry dataset, with each record being given the value 1, before the aggregation was carried out. (Some

versions of SPSS will create a 'count' variable as a matter of course.) The second variable, the key stage point score variable, in the 'Summaries of Variable(s):' section has been left in the default position. This will provide the average examination point score for each candidate.

Unusually, SPSS allows Aggregate\Summaries of Variables function to use same variable more than once in the same exercise. The third variable shows that the point score variable is to be used again, with the function changed to Sum. This will give each candidate's total point score. The 'Aggregate Data' dialogue box also has a 'Name and Label' button to the right of the 'Function' button, as shown in Figure 82. Selecting a variable in the 'Summaries of Variable(s):', and then selecting 'Name and Label' allow you to change a variable's name and to give it a label to add meaning to output.

Figure 83. Choosing aggregate functions



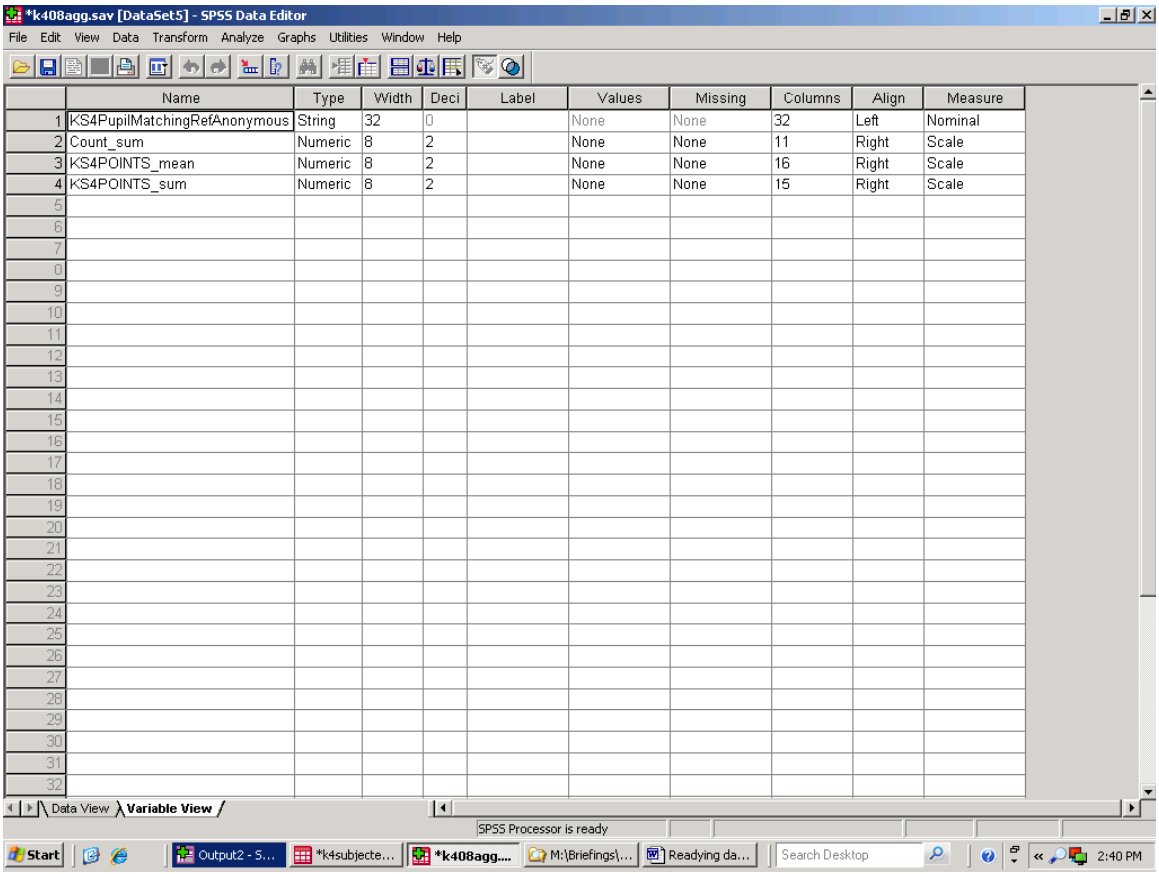
Selecting 'Continue' in the 'Aggregate Data: Aggregate Function' window returns the user to the 'Aggregate Data' dialogue box. At this stage, or before the aggregate functions are chosen, users can specify where the file is to go, and what it will be called by selecting the 'File' button in the lower part of the 'Aggregate Data' dialogue box.

At least at the outset, choose a meaningful file name at this point. The default position gives an aggregated filed the name agg.sav. If an agg.sav file already exists in the directory you have chosen, it will be over-written and this can create real problems if that file is still needed.

Since the NPD files are large, 'Sort before aggregating' has been selected at the bottom of the Aggregate Data dialogue box. Once the options needed have been chosen, selecting 'OK' in the 'Aggregate Data' dialogue box sets the procedure in motion.

Figure 84 shows the SPSS Variable View of the aggregated dataset. The same dataset in SPSS Data View would show that the two totals and the one average have also been calculated. The aggregate dataset also includes the break variable, which is the code for the individual candidate. This can be used to add the pupil totals and the pupil average figures to the wider pupil dataset using the procedures illustrated in earlier Sections.

Figure 84. The aggregated dataset in SPSS Variable View



23. Grouping frequently used variables in sets to reduce the time needed to locate variables in large datasets

Merging files and creating derived variables can produce large datasets. The Appendix to the Guide shows that the Merged 2002 to 2005 Pupil dataset has 1,200+ variables. With large datasets, simply locating the variables needed for analysis can time-consuming and, frankly, irritating. If the same subset of variables will be used on a number of occasions, you can group these as a 'set' which can then be called up quickly and easily for future use. When a particular set is called up, only those variables listed in it will be shown and that should simplify finding the variables need. This procedure does not create yet another space hungry dataset. The set remains a part of the wider dataset on which it draws.

To establish a set, select 'Utilities' from the SPSS main menu, and then select 'Define Sets' from the dropdown list. In this instance, assume that we wish to analyse pupil entitlement to free school meals, in the light of pupil characteristics such as ethnicity, gender and whether the child has ever been in care while on roll at the current school.

Figure 85. Utilities and Define Sets

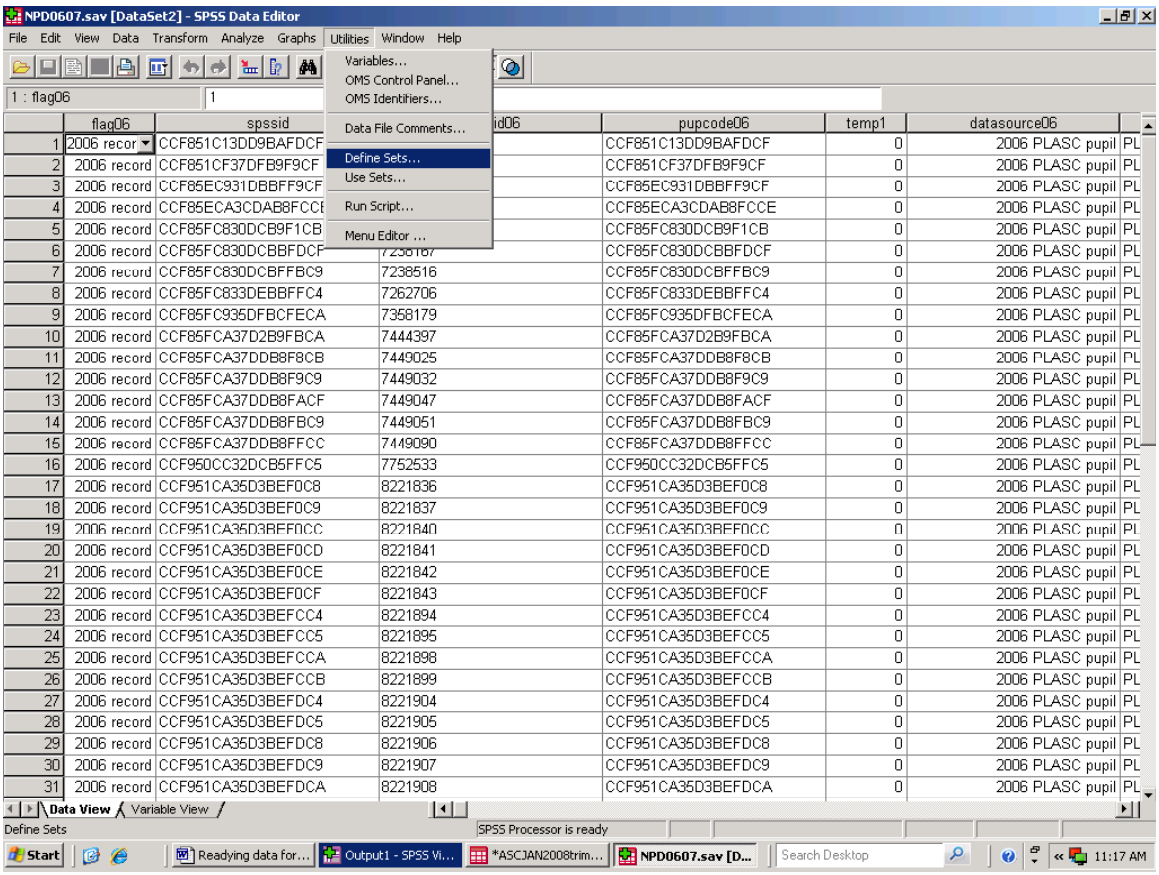


Figure 87 shows the Define Variable Sets dialogue window with the full list of variables in the dataset on the left. A selection of these have been transferred to the 'Variables in Set' section to the right, and the set has been given the name 'FSM', The 'Add' button has already been selected, and the set has been added to other, already existing, SPSS variable sets.

To bring a variable set into play, select 'Utilities' from the SPSS main menu, followed by 'Use Sets' from the dropdown list and this will open the window shown in Figure 88. The 'FSM' set will be listed on the left. Use the arrow immediately to the right of the list of variables to transfer the 'FSM' set to the pane to the right. If you now invoke a simple SPSS statistical exercise, such as running a Table, where you are presented with a list of variables to choose from you will, and contrary to what you wanted, be presented with the full list of variables in the dataset, and not the subset you just established as the FSM set.

What has gone wrong? Figure 88 shows that there are also other sets in the 'Sets in Use' pane, including 'ALL VARIABLES' so, yes, SPSS will show all variables. Transfer the sets that you do not intend to use back to the pane on the left of the 'Use Sets' dialogue box, and all will be well.

Figure 86. Selecting the variables to be included in the set and naming the set

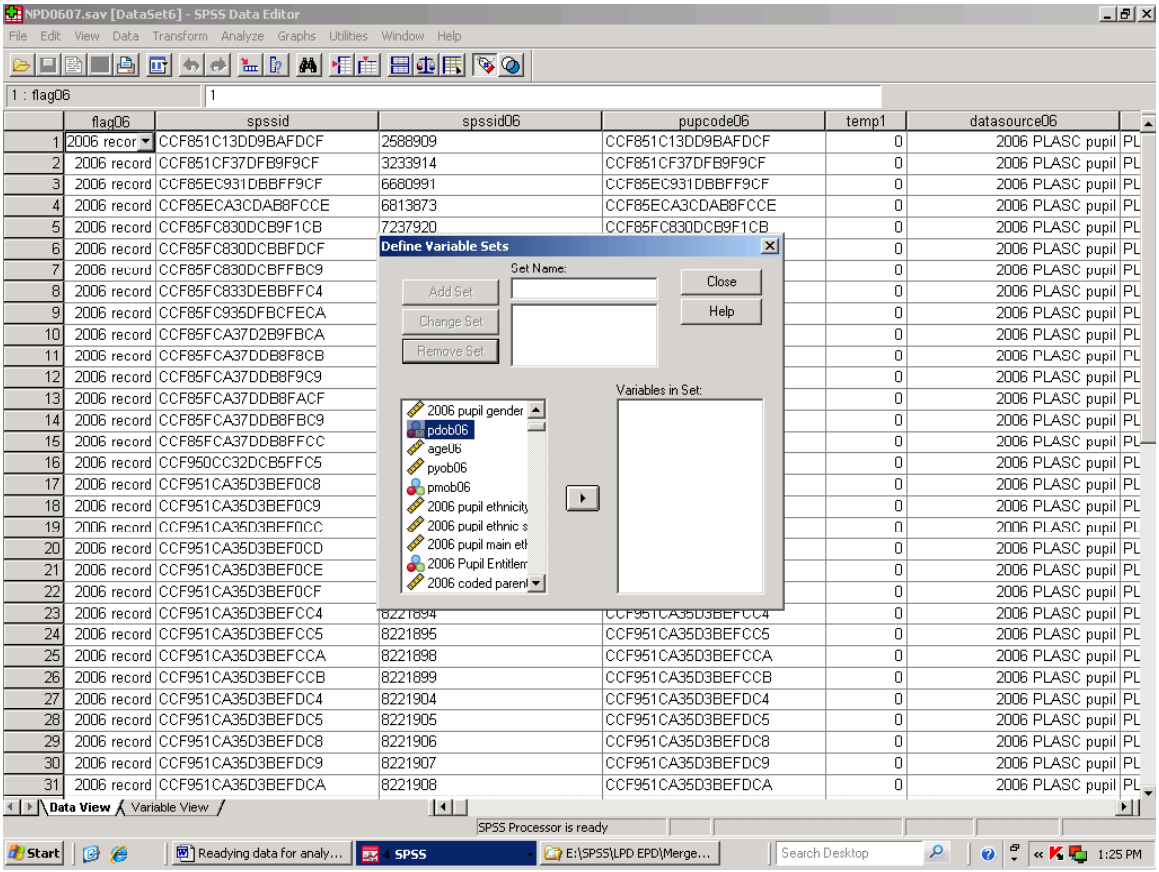


Figure 87. Selecting variables to be included in a set

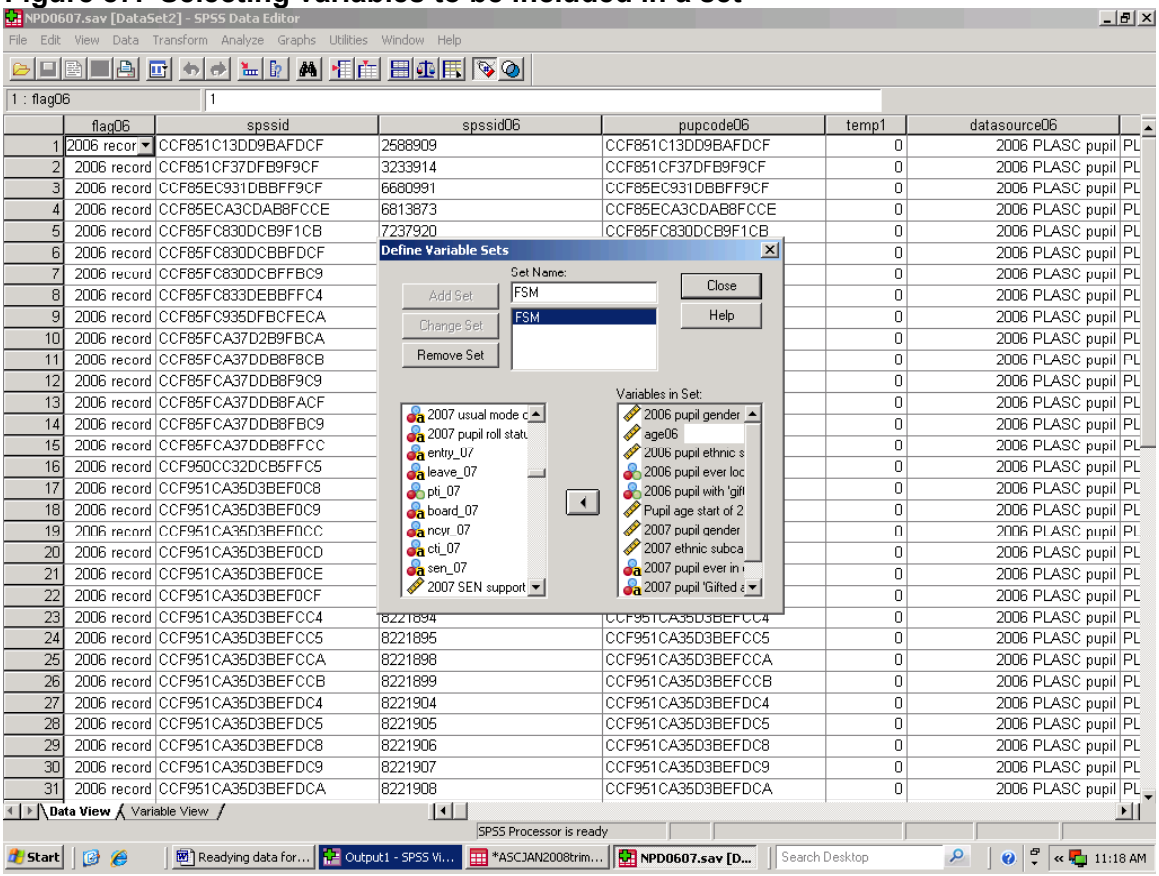


Figure 88. Transferring the new set from ‘Use Sets’ to ‘Sets in Use’

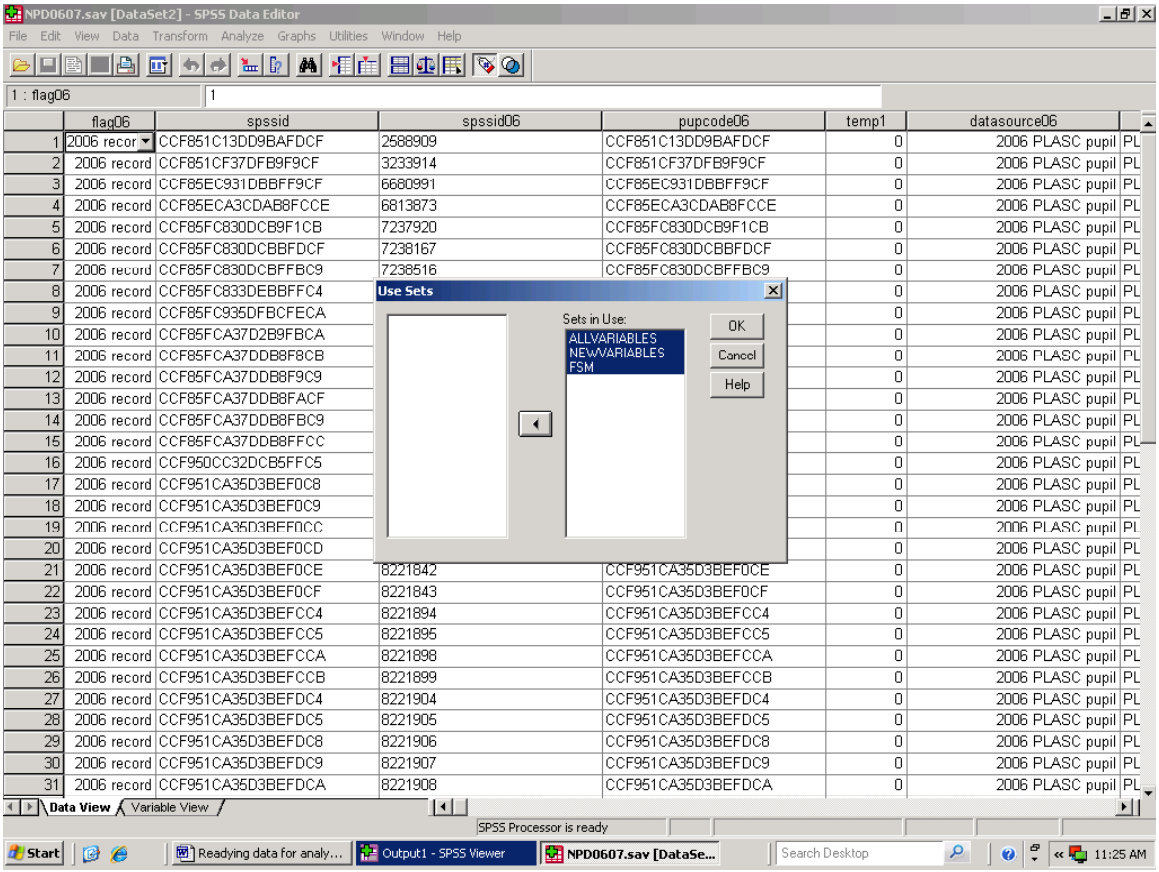
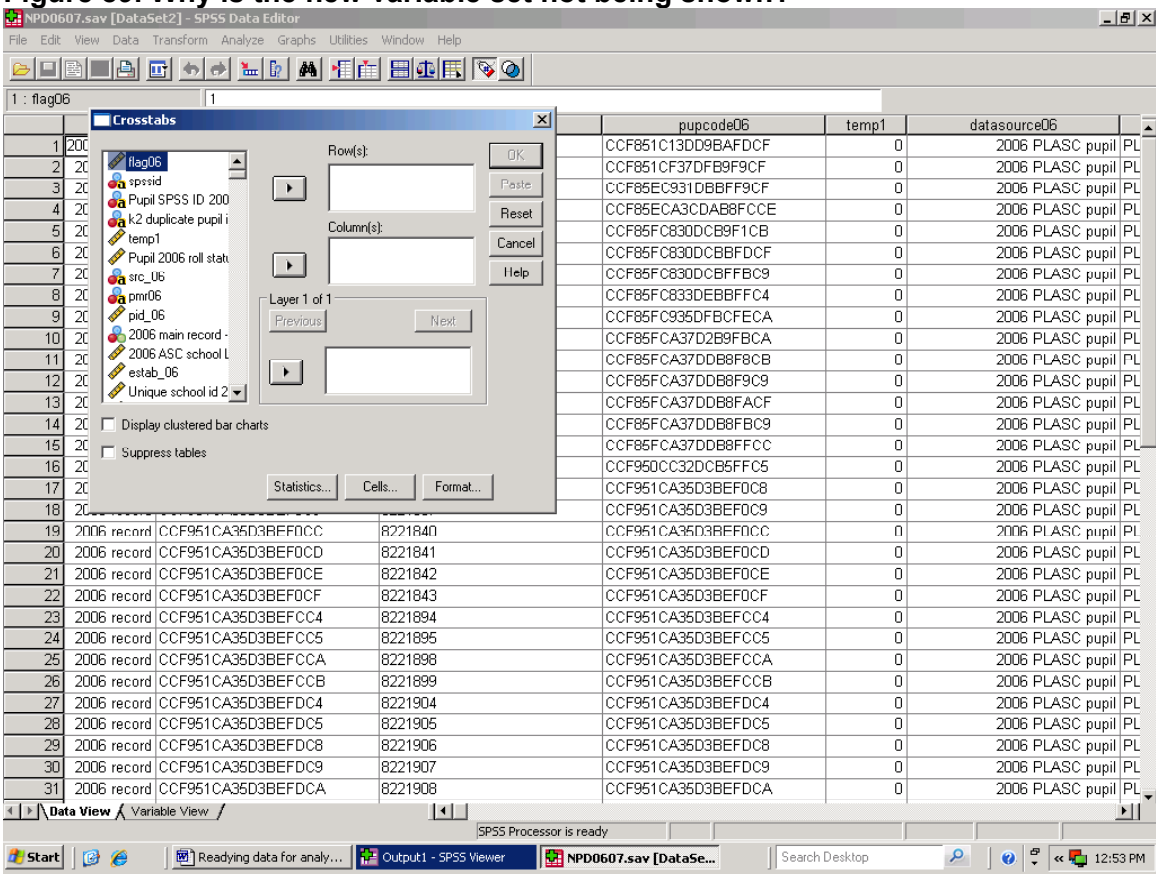


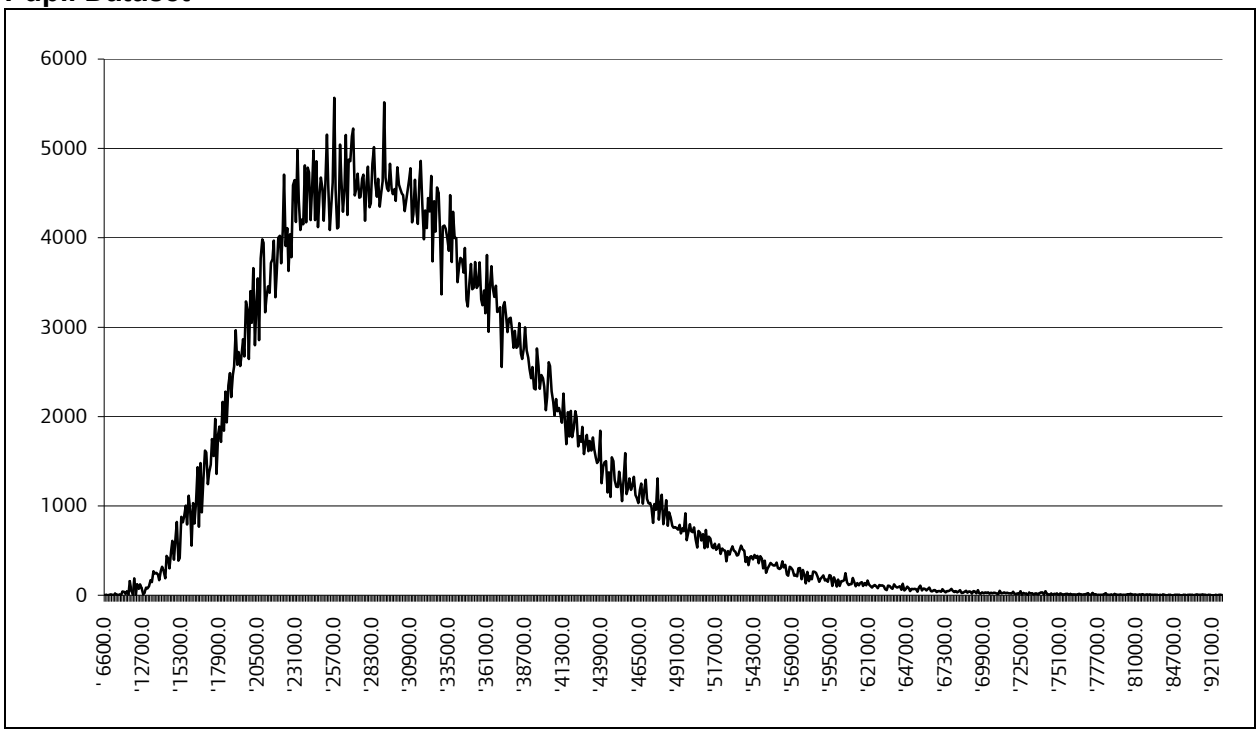
Figure 89. Why is the new variable set not being shown?



24. Grouping values in a derived variable for analytical purposes – Visual Bander

PayCheck equivalised income at postcode level was added to the 2004 London Pupil Dataset to provide a measure of the level of pupil social advantage and disadvantage. Figure 90 shows that the distribution of that data in the 2004 LPD is reassuringly close to a Bell curve. Once added to the LPD, the data were also grouped into five income categories, and one aim in doing this was to ensure that there were enough pupil records in the *intermediate* income category to support sensible comparisons with pupils living in a high income group area and living in the lowest income group area.

Figure 90. Distribution of pupils by equivalised income of home postcode. 2004 London Pupil Dataset



Source: 2004 LPD

The boundaries of the five groups could have been established in the light of information in an equivalised income Frequency Table. Set up, but do not run, a frequency table for the variable in question in the ordinary way. Now click on the 'Statistics' button in the 'Frequencies dialogue box, as shown in Figure 91. This allows cut points to be established for a user-specified number of groups, and where those fall will be shown in the Frequency Table output. This approach does not create a new banded equivalised income variable, but it does establish where band limits fall, and these can then be used to create a new banded variable using the Recode or Compute procedures.

Figure 92 illustrates a further grouped variable. In this example, five groups were established based on level of attainment in 2006 key stage 2 writing tests, and using SPSS Visual Bander. The Figure includes a separate 6th group, which has been created separately and shows pupils who have no key stage 2 writing test result. The Figure also shows average home to school distance in each of the attainment quintiles. That distance increases with higher levels of attainment, except that those who lack an assessment record appear to be travelling furthest to reach school.

The SPSS Visual Bander facility allows an additional banded variable to be created directly within the dataset, without recourse to the Recode or Compute procedures. Visual bander also provides for user-defined limits to bands. The starting point is the 'Transform' option in the SPSS main window as shown in Figure 93. This produces the drop down list shown in that Figure.

Selecting 'Visual bander' from that list produces a list of variables as in Figure 94. What in SPSS terms are scale or ordinal values are the only values to be shown; there is little or no point in using Visual Bander to group nominal data. If you wish to band a numeric variable, and it is not shown in the window, open the Variable View window on the dataset, and check whether the variable is listed as a nominal variable in the last column to the right (the 'Measure' column). If it is, click on 'Nominal' and select 'Scale' or 'Ordinal' depending on the type of data to be banded.

Additionally, this procedure will ignore missing values where these are not coded as such. How that is dealt with depends on the research question being raised, but as Figure 92 illustrates, cases with missing values may well be of interest in themselves in some projects.

Figure 91. Establishing banded cut off points to be printed in a Frequency Table

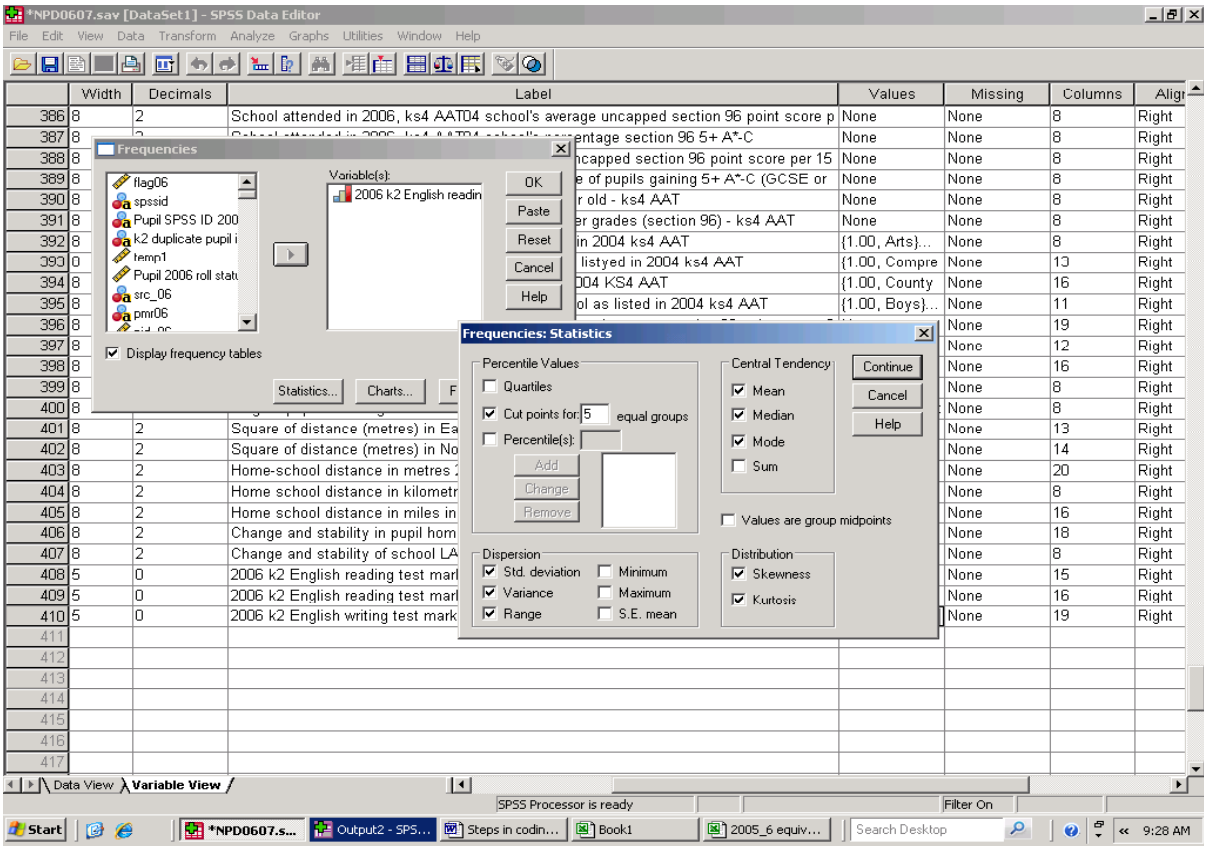
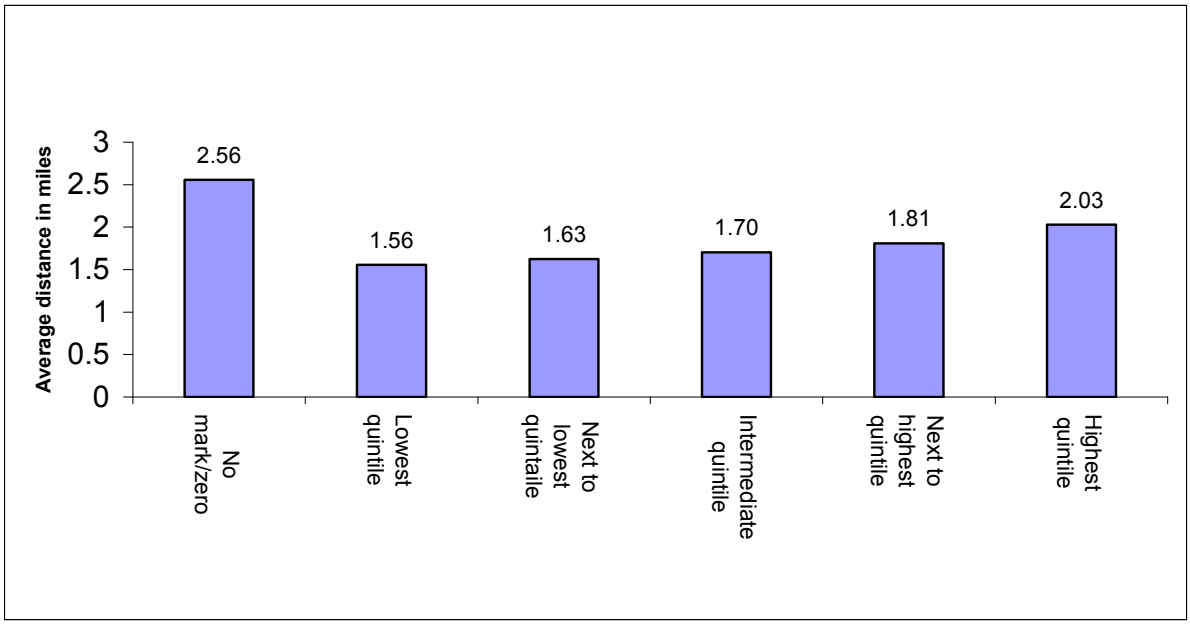
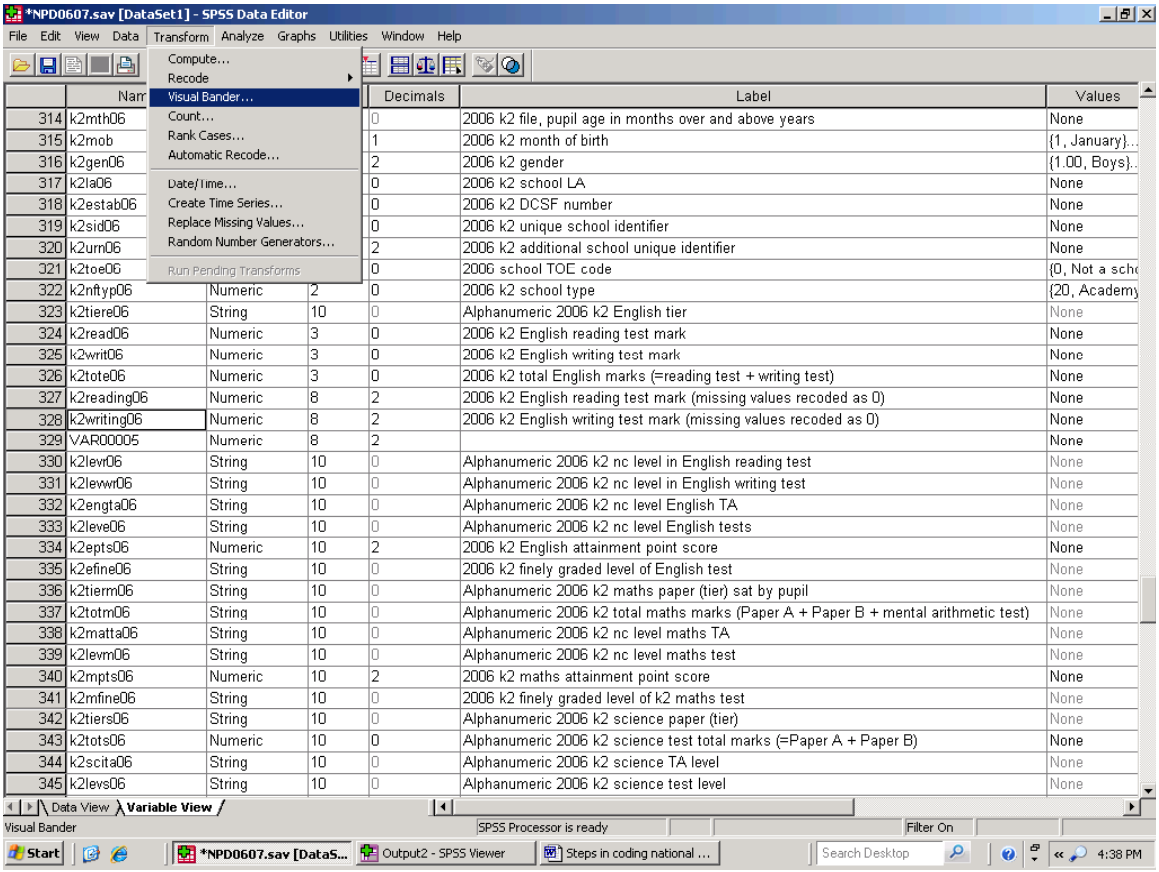


Figure 92. Home-school straight line distance in 2007 for pupils aged 10 in 2006, by key stage 2 writing test point score quintiles



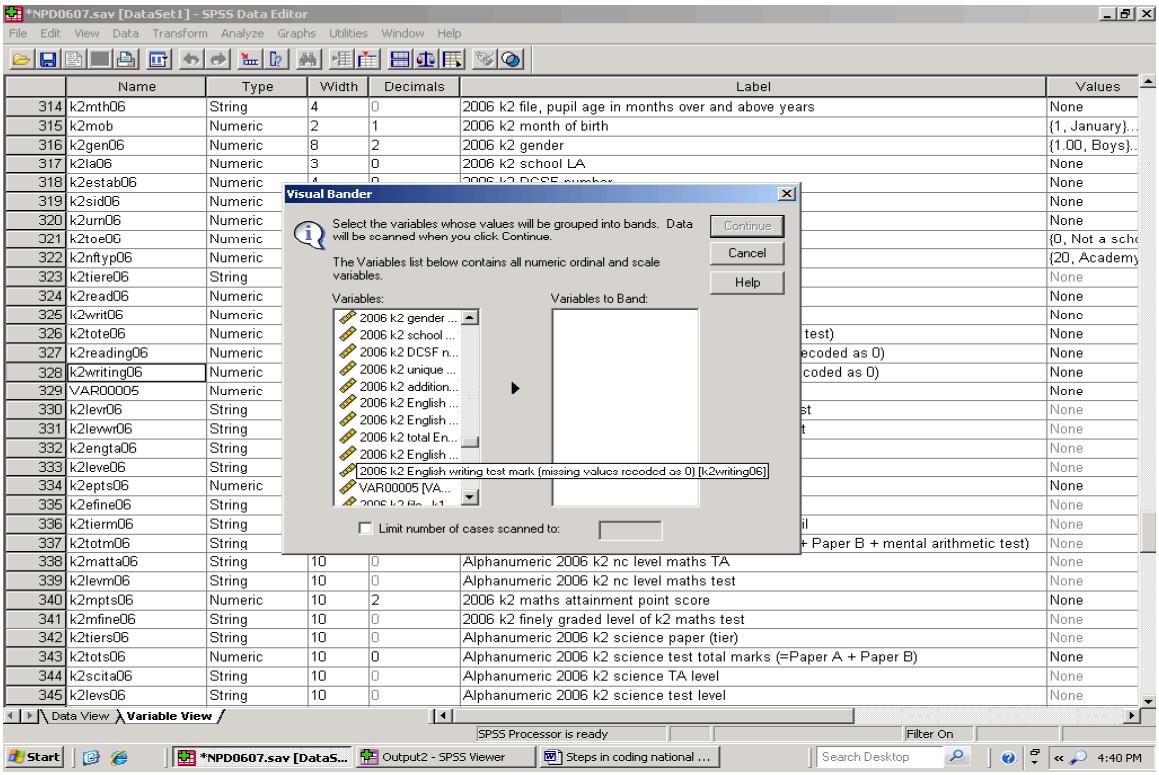
Source: merged 2006 2007 English Pupil Dataset

Figure 93. The first step in using Visual bander – Transform\Visual Bander



	Name	Type	Width	Decimals	Label	Values
314	k2mth06	String	4	0	2006 k2 file, pupil age in months over and above years	None
315	k2mob	Numeric	2	1	2006 k2 month of birth	{1, January}...
316	k2gen06	Numeric	8	2	2006 k2 gender	{1.00, Boys}...
317	k2la06	Numeric	3	0	2006 k2 school LA	None
318	k2estab06	Numeric	4	0	2006 k2 DCSF number	None
319	k2sid06	Numeric	8	0	2006 k2 unique school identifier	None
320	k2urn06	Numeric	6	2	2006 k2 additional school unique identifier	None
321	k2toe06	Numeric	3	0	2006 school TOE code	{0, Not a sch
322	k2nftyp06	Numeric	2	0	2006 k2 school type	{20, Academy
323	k2tiere06	String	10	0	Alphanumeric 2006 k2 English tier	None
324	k2read06	Numeric	3	0	2006 k2 English reading test mark	None
325	k2writ06	Numeric	3	0	2006 k2 English writing test mark	None
326	k2tote06	Numeric	3	0	2006 k2 total English marks (=reading test + writing test)	None
327	k2reading06	Numeric	8	2	2006 k2 English reading test mark (missing values recoded as 0)	None
328	k2writing06	Numeric	8	2	2006 k2 English writing test mark (missing values recoded as 0)	None
329	VAR00005	Numeric	8	2		None
330	k2levr06	String	10	0	Alphanumeric 2006 k2 nc level in English reading test	None
331	k2levwr06	String	10	0	Alphanumeric 2006 k2 nc level in English writing test	None
332	k2engta06	String	10	0	Alphanumeric 2006 k2 nc level English TA	None
333	k2leve06	String	10	0	Alphanumeric 2006 k2 nc level English tests	None
334	k2epts06	Numeric	10	2	2006 k2 English attainment point score	None
335	k2efine06	String	10	0	2006 k2 finely graded level of English test	None
336	k2tierm06	String	10	0	Alphanumeric 2006 k2 maths paper (tier) sat by pupil	None
337	k2totm06	String	10	0	Alphanumeric 2006 k2 total maths marks (Paper A + Paper B + mental arithmetic test)	None
338	k2matta06	String	10	0	Alphanumeric 2006 k2 nc level maths TA	None
339	k2levm06	String	10	0	Alphanumeric 2006 k2 nc level maths test	None
340	k2mpts06	Numeric	10	2	2006 k2 maths attainment point score	None
341	k2mfine06	String	10	0	2006 k2 finely graded level of k2 maths test	None
342	k2tiers06	String	10	0	Alphanumeric 2006 k2 science paper (tier)	None
343	k2tots06	Numeric	10	0	Alphanumeric 2006 k2 science test total marks (=Paper A + Paper B)	None
344	k2scita06	String	10	0	Alphanumeric 2006 k2 science TA level	None
345	k2levs06	String	10	0	Alphanumeric 2006 k2 science test level	None

Figure 94. Selecting variables to be grouped as banded variables



Having selected 'Transform' and then 'visual Bander', and all being well, the variable/s you are interested in will appear in the list on the left of the 'Visual Bander' dialogue box. Select the variable of interest, and click on the 'arrow' button to the right to transfer the variable name to the 'Variables to Band' section of the Visual Bander dialogue box, and then select the 'Continue'

button. This will produce a pop up window labelled 'Visual Bander', with the message 'Scanning Places. Please wait' and a 'clock' showing the number of cases counted. With large datasets, this phase can be time-consuming, though not as time-consuming as a data 'Sort Cases' exercise.

When the count is complete, SPSS will provide the 'Visual Bander' dialogue box shown in Figure 95 (though the 'Make Cutpoints' dialogue box will not be shown at this stage).

The variable previously selected will be shown in the 'Scanned Variable List' section of a new version of the 'Visual Bander' dialogue box. Left clicking on that variable transfers its name and label (if any) to the Current Variable 'Name' and 'Label' sections of the dialogue box. The user can then type in a name and label for the new banded variable in the Banded Variable section of the dialogue box.

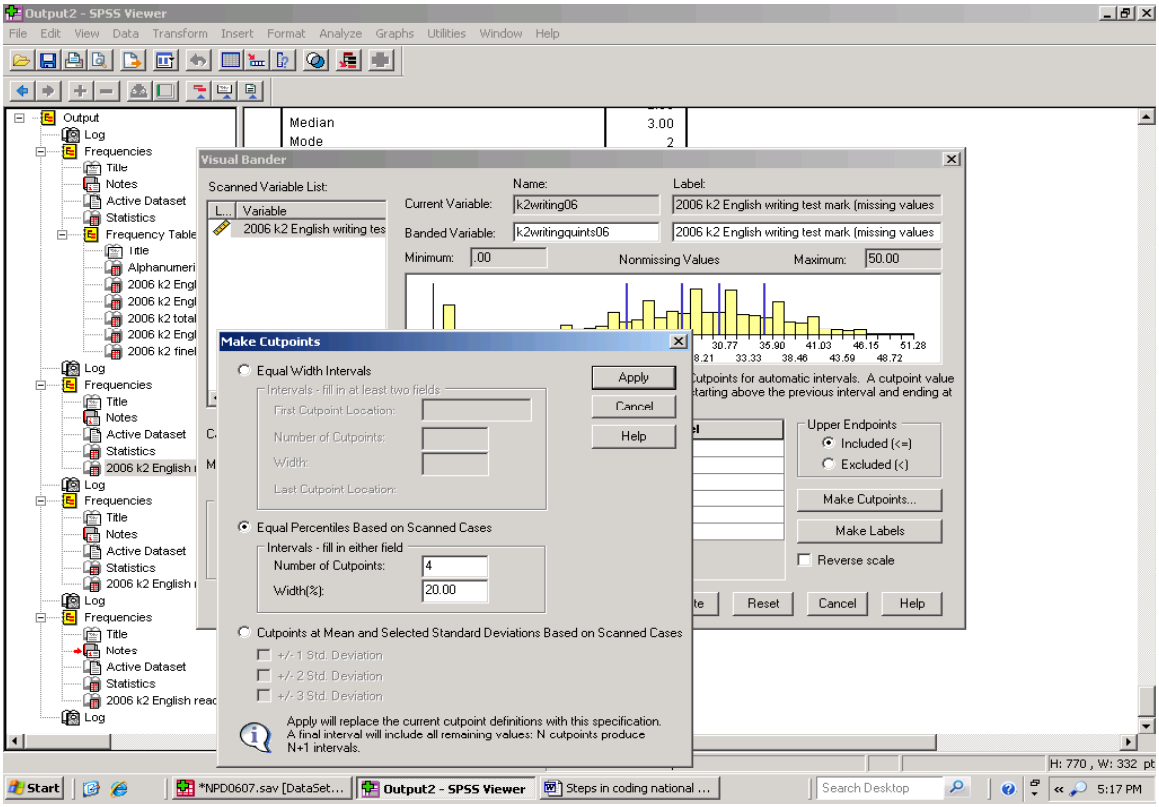
Selecting the 'Make Cutpoints' button, in the lower right of the Visual Bander dialogue box, produces the 'Make Cutpoints' dialogue box shown in Figure 95. In this example, 4 cut points have been selected, and the 'Equal Percentiles' radio button has also been selected. This will produced five groups, each with 20 per cent of the key stage cohort. Once the choices in the 'Make Cut points' dialogue box have been made, select 'apply' in the top right of the dialogue box.

The 'Visual Bander' window, shown in Figures 95 and 96, gives a histogram for the data, and shows where the Cutpoints will fall. Depending on the nature of your data, and any planning that may have taken place in advance, you may now be well advised to stop and consider whether the groups indicated are really what you need. If you are dealing with income data, are those with the highest income groups identified separately, or are they submerged within a wider income band?

In the case of key stage 2 writing score, there is a clear bulge at the lowest end of the range. Should that group be identified separately, or should it be left within a broader low raw score attainment range? DMAG Briefing 2005 – 32 (Moving home and changing school – 1) and DMAG Briefing 2008 – 27 (Social Selection, social Sorting and Education – 2: 'Missing' children) provide the answer in this case. 'The literature', as well as specific research questions, should inform where cutpoints fall.

In the Visual Bander dialogue box, users can click on each cutpoint and move them if the research questions you are seeking to answer suggest that would be appropriate. (You are advised to run a frequency table at a later point to determine whether any new bands contain a sufficient number of cases).

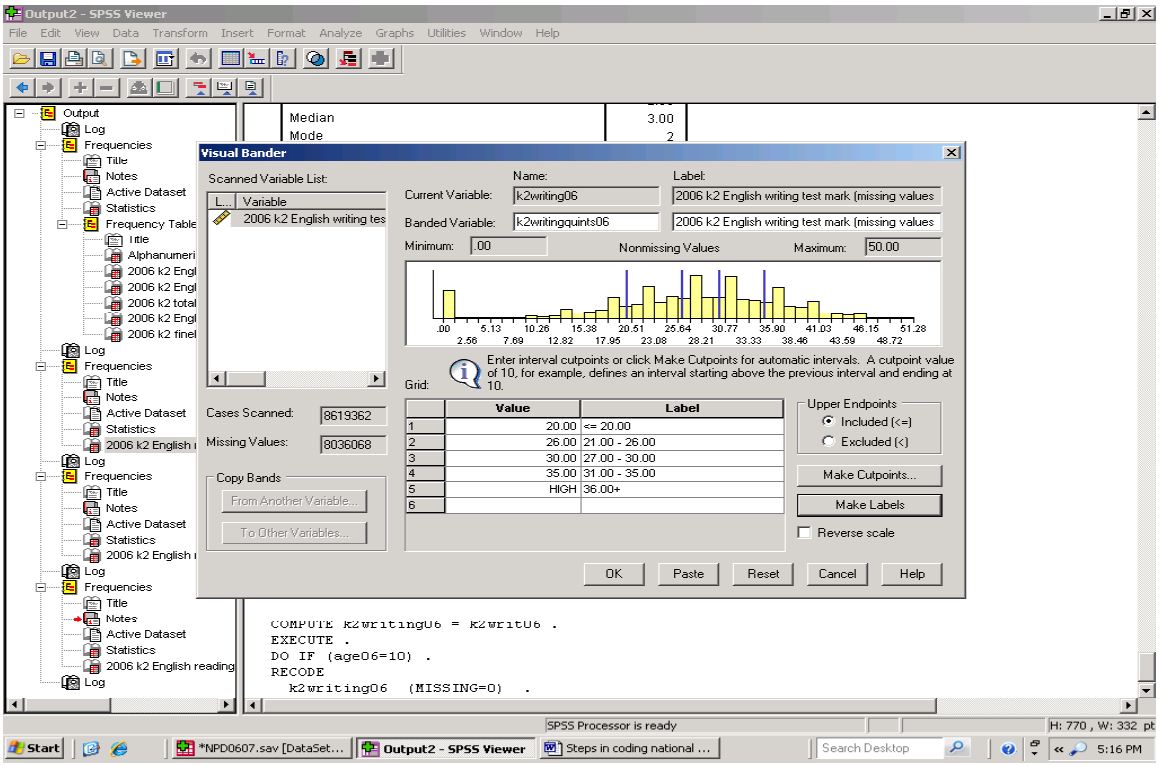
Figure 95. Selecting Cut off points in Visual Bander



Clicking on ‘Make Labels’ in the Visual Bander dialogue box allows the user to create value labels for each band, as shown in Figure 96. Value labels add meaning to a variable which, in this example, would otherwise be listed as 1 to 5 (Is 1 the high income/high raw score attainment group, or is it 5?) when the value labels have been typed in, click on the ‘OK’ near the lower edge of the dialogue box, and the new banded variable will be created.

Using Visual is quicker than the ‘Frequency Tables plus Recode or Compute’ approach outlined at the beginning of this section, and this will be welcome news to some. However, whether this derived variable is to be used to simplify data for a particular audience, or whether there is a specific analytical issue at stake, the number and placement of cut points, and the number of cases in each group, should be considered carefully and in advance.

Figure 96. Selecting Labels to identify groups, plus the long tail of underachievement



25. Syntax files

If a particular set of procedures is used on several occasions, it may be possible to automate them using a syntax file. At one level, a syntax file is simple a record kept by SPSS of the procedures files and so on that a research analyst has worked with, and the text below Figure 31 in Section 10 is part of a record of that type.

Additionally, there are instances where a Windows 'point and click' approach in SPSS will not work, and an alternative will need to be found. Figure 97 illustrates part of the SPSS record of an attempt to recode string information on pupil mode of travel to school into a different numeric variable. The approach is the same as that illustrated earlier in the Guide. With 21 codes to work with SPSS needed more than one line to record the work, with each line being separated by an appropriate character. In this instance, the Windows, 'point and click' approach to listing old and new values did not include the line separators needed and the procedure failed. One option would be to recode string codes in smaller batches of (say) six codes at a time. A further option is to consider the potential value of syntax files.

Figure 97. Windows 'point and click' will not always work with SPSS

```
RECODE
  pModeOfTravel08
  (MISSING=14)  ('BDR'=12)  ('BNK'=7)  ('BUS'=7)  ('CAR'=3)  ('CRS'=4)
('C'+
 'YC'=2)  ('CYCLE'=2)  ('DSB'=6)  ('LEFT'=14)  ('LUL'=10)  ('MTL'=11)  ('OT

>Warning # 208 in column 73.  Text: OT
>A text string is not correctly enclosed in quotation marks on the command
>line.  Literals may not be continued across command lines without the use
>of the continuation symbol '+'.

 H'=13)  ('OTHER'=13)  ('PBS'=5)  ('PSB'=5)  ('TAXI'=8)  ('TRAIN'=9)
('TRN''+
 '=9)  ('TXI'=8)  ('WALK'=1)  ('WLK'=1)  INTO  ptraveltoschl08 .

>Warning # 208 in column 36.  Text: =1)  INTO  ptraveltoschl08
>A text string is not correctly enclosed in quotation marks on the command
>line.  Literals may not be continued across command lines without the use
>of the continuation symbol '+'.
```

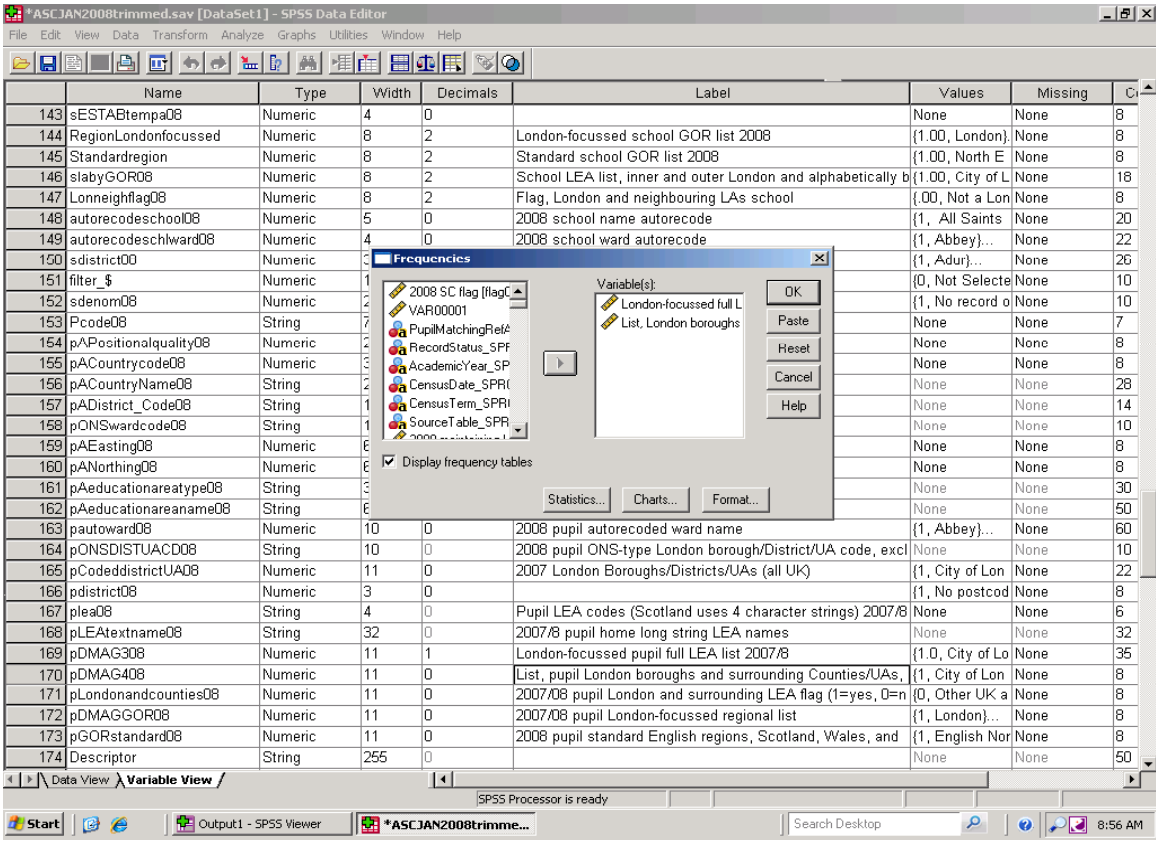
For those new to them, syntax files may look like computer programs, and advanced work with them is something of a specialist activity. (The Syntax Guide to SPSS 14 is 2,079 pages long). The Syntax Guide aside, there is a way of beginning work with syntax files which may be familiar to those who have worked with spreadsheets. This involves using the SPSS record of your actions, and pasting that into a Syntax window, in much the same way that spreadsheet software can automatically record in a macro each step taken by the user.

The example shown here illustrates work to identify the number of pupils who attend a school maintained by their home local authority area, and the number who attend a school maintained by a local authority other than the one in whose area they live. In plain English, this would mean for example in the case of pupils living in Hertfordshire, the number who attend a school maintained by Hertfordshire County Council and the number attending a school maintained by another local authority. All other things being equal, a simple comparison of a school's three digit local authority code would provide the answer needed. In London parlance, pupils whose home LA area code is the same as the LA code of the school attended, attend 'in-borough' schools. A pupil with a home LA code which differs from the LA code of the school attended is on roll in an 'out-borough' school. However, before comparing LA codes, it would be prudent to run Frequency Tables for pupil home LA area code, and for the LA code of the school attended, to check for missing data and any other anomalies.

Figure 98 shows the familiar Frequencies dialogue box, in this case with two variables selected. Each identifies a local authority code, and there is a 'Paste' button in the dialogue box to the right of the second of the two variables. Selecting the Paste button opens a Syntax window, as illustrated in Figure 99 which, in SPSS-speak, shows the command to run the two Frequency Tables.

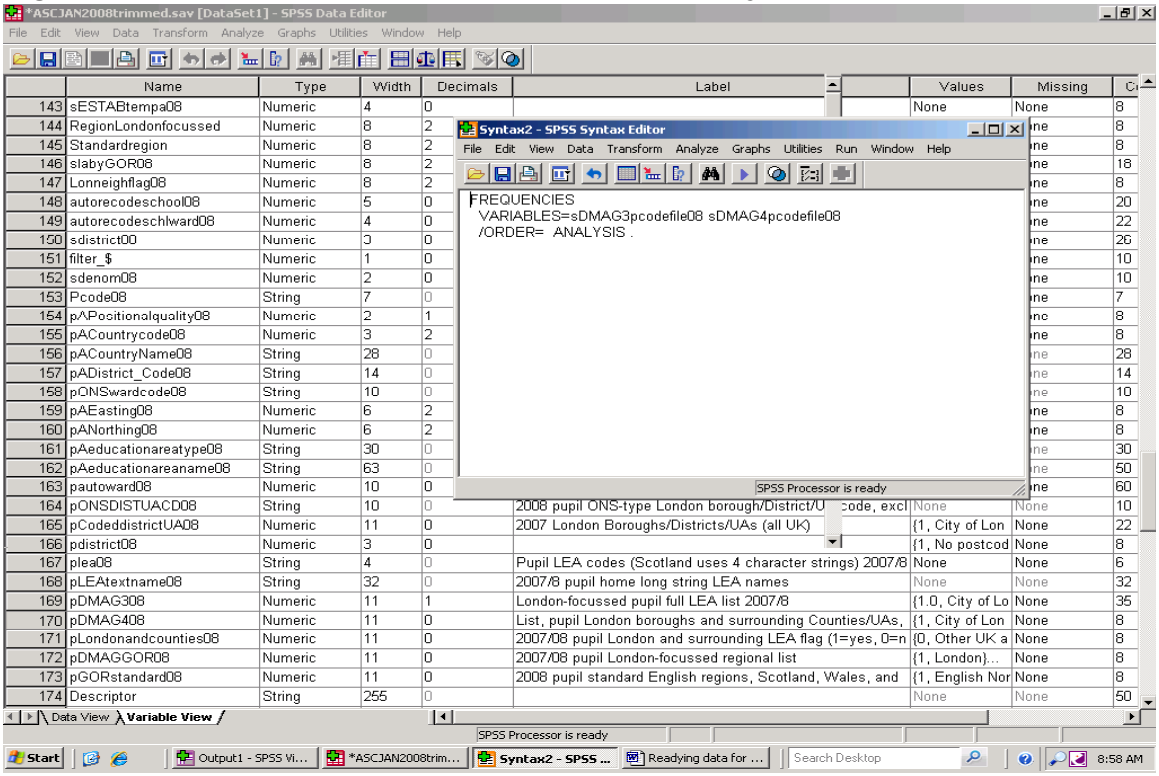
The SPSS 'Syntax – SPSS Syntax Editor' window has a menu along the top, which includes 'Run'. Selecting 'Run' followed by 'All', as shown in Figure 100, sets that command in motion.

Figure 98. The Paste button in the Frequencies window



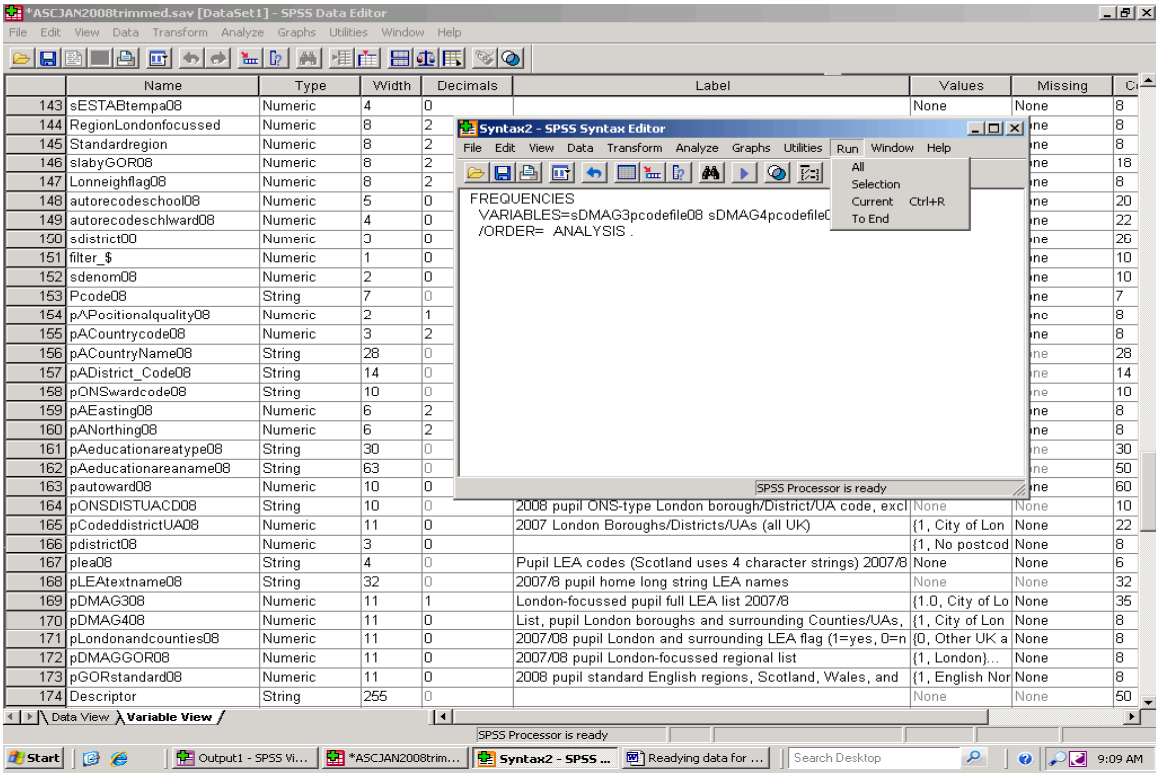
	Name	Type	Width	Decimals	Label	Values	Missing	C
143	sESTABtempa08	Numeric	4	0		None	None	8
144	RegionLondonfocussed	Numeric	8	2	London-focussed school GOR list 2008	{1.00, London}.	None	8
145	Standardregion	Numeric	8	2	Standard school GOR list 2008	{1.00, North E	None	8
146	slabyGOR08	Numeric	8	2	School LEA list, inner and outer London and alphabetically b	{1.00, City of L	None	18
147	Lonneighflag08	Numeric	8	2	Flag, London and neighbouring LAs school	{.00, Not a Lon	None	8
148	autorecodeschool08	Numeric	5	0	2008 school name autorecode	{1, All Saints	None	20
149	autorecodeschlward08	Numeric	4	0	2008 school ward autorecode	{1, Abbey}...	None	22
150	sdistrict00	Numeric	3	0		{1, Adur}...	None	26
151	filter_$	Numeric	1	0		{0, Not Selecte	None	10
152	sdenom08	Numeric	2	0		{1, No record o	None	10
153	Pcode08	String	7	0		None	None	7
154	pAPositionalquality08	Numeric	2	1		None	None	8
155	pACountrycode08	Numeric	3	2		None	None	8
156	pACountryName08	String	28	0		None	None	28
157	pADistrict_Code08	String	14	0		None	None	14
158	pONSwardcode08	String	10	0		None	None	10
159	pAEasting08	Numeric	6	2		None	None	8
160	pANorthing08	Numeric	6	2		None	None	8
161	pAeducationareatype08	String	30	0		None	None	30
162	pAeducationareaname08	String	63	0		None	None	50
163	pautoward08	Numeric	10	0	2008 pupil autorecoded ward name	{1, Abbey}...	None	60
164	pONSDISTUACD08	String	10	0	2008 pupil ONS-type London borough/District/UA code, excl	None	None	10
165	pCodeddistrictUA08	Numeric	11	0	2007 London Boroughs/Districts/UAs (all UK)	{1, City of Lon	None	22
166	pdistrict08	Numeric	3	0		{1, No postcod	None	8
167	plea08	String	4	0	Pupil LEA codes (Scotland uses 4 character strings) 2007/8	None	None	6
168	pLEAtextname08	String	32	0	2007/8 pupil home long string LEA names	None	None	32
169	pDMAG308	Numeric	11	1	London-focussed pupil full LEA list 2007/8	{1.0, City of Lo	None	35
170	pDMAG408	Numeric	11	0	List, pupil London boroughs and surrounding Counties/UAs,	{1, City of Lon	None	8
171	pLondonandcounties08	Numeric	11	0	2007/08 pupil London and surrounding LEA flag (1=yes, 0=n	{0, Other UK a	None	8
172	pDMAGGOR08	Numeric	11	0	2007/08 pupil London-focussed regional list	{1, London}...	None	8
173	pGORstandard08	Numeric	11	0	2008 pupil standard English regions, Scotland, Wales, and	{1, English Nor	None	8
174	Descriptor	String	255	0		None	None	50

Figure 99. Press the 'Paste' button, and there is the Syntax window



	Name	Type	Width	Decimals	Label	Values	Missing	C
143	sESTABtempa08	Numeric	4	0		None	None	8
144	RegionLondonfocussed	Numeric	8	2			ne	8
145	Standardregion	Numeric	8	2			ne	8
146	slabyGOR08	Numeric	8	2			ne	18
147	Lonneighflag08	Numeric	8	2			ne	8
148	autorecodeschool08	Numeric	5	0			ne	20
149	autorecodeschlward08	Numeric	4	0			ne	22
150	sdistrict00	Numeric	3	0			ne	26
151	filter_$	Numeric	1	0			ne	10
152	sdenom08	Numeric	2	0			ne	10
153	Pcode08	String	7	0			ne	7
154	pAPositionalquality08	Numeric	2	1			ne	8
155	pACountrycode08	Numeric	3	2			ne	8
156	pACountryName08	String	28	0			ne	28
157	pADistrict_Code08	String	14	0			ne	14
158	pONSwardcode08	String	10	0			ne	10
159	pAEasting08	Numeric	6	2			ne	8
160	pANorthing08	Numeric	6	2			ne	8
161	pAeducationareatype08	String	30	0			ne	30
162	pAeducationareaname08	String	63	0			ne	50
163	pautoward08	Numeric	10	0			ne	60
164	pONSDISTUACD08	String	10	0	2008 pupil ONS-type London borough/District/U	code, excl	None	10
165	pCodeddistrictUA08	Numeric	11	0	2007 London Boroughs/Districts/UAs (all UK)	{1, City of Lon	None	22
166	pdistrict08	Numeric	3	0		{1, No postcod	None	8
167	plea08	String	4	0	Pupil LEA codes (Scotland uses 4 character strings) 2007/8	None	None	6
168	pLEAtextname08	String	32	0	2007/8 pupil home long string LEA names	None	None	32
169	pDMAG308	Numeric	11	1	London-focussed pupil full LEA list 2007/8	{1.0, City of Lo	None	35
170	pDMAG408	Numeric	11	0	List, pupil London boroughs and surrounding Counties/UAs,	{1, City of Lon	None	8
171	pLondonandcounties08	Numeric	11	0	2007/08 pupil London and surrounding LEA flag (1=yes, 0=n	{0, Other UK a	None	8
172	pDMAGGOR08	Numeric	11	0	2007/08 pupil London-focussed regional list	{1, London}...	None	8
173	pGORstandard08	Numeric	11	0	2008 pupil standard English regions, Scotland, Wales, and	{1, English Nor	None	8
174	Descriptor	String	255	0		None	None	50

Figure 100. Select 'Run' and 'All' in the Syntax window



	Name	Type	Width	Decimals	Label	Values	Missing
143	sESTABtempa08	Numeric	4	0		None	None
144	RegionLondonfocussed	Numeric	8	2			
145	Standardregion	Numeric	8	2			
146	slabyGOR08	Numeric	8	2			
147	Lonneighflag08	Numeric	8	2			
148	autorecodeschool08	Numeric	5	0			
149	autorecodeschlward08	Numeric	4	0			
150	sdistrict00	Numeric	3	0			
151	filter_$	Numeric	1	0			
152	sdenom08	Numeric	2	0			
153	Pcode08	String	7	0			
154	pAPositionalquality08	Numeric	2	1			
155	pACountrycode08	Numeric	3	2			
156	pACountryName08	String	28	0			
157	pADistrict_Code08	String	14	0			
158	pONSwardcode08	String	10	0			
159	pAEasting08	Numeric	6	2			
160	pANorthing08	Numeric	6	2			
161	pAeducationareatype08	String	30	0			
162	pAeducationareaname08	String	63	0			
163	pautoward08	Numeric	10	0			
164	pONSDISTUACD08	String	10	0	2008 pupil ONS-type London borough/District/UA code, excl	None	None
165	pCodeddistrictUA08	Numeric	11	0	2007 London Boroughs/Districts/UAs (all UK)	{1, City of Lon	None
166	pdistrict08	Numeric	3	0		{1, No postcod	None
167	plea08	String	4	0	Pupil LEA codes (Scotland uses 4 character strings) 2007/8	None	None
168	pLEAtextname08	String	32	0	2007/8 pupil home long string LEA names	None	None
169	pDMAG308	Numeric	11	1	London-focussed pupil full LEA list 2007/8	{1.0, City of Lo	None
170	pDMAG408	Numeric	11	0	List, pupil London boroughs and surrounding Counties/UAs,	{1, City of Lon	None
171	pLondonandcounties08	Numeric	11	0	2007/08 pupil London and surrounding LEA flag (1=yes, 0=n	{0, Other UK a	None
172	pDMAGGOR08	Numeric	11	0	2007/08 pupil London-focussed regional list	{1, London}...	None
173	pGORstandard08	Numeric	11	0	2008 pupil standard English regions, Scotland, Wales, and	{1, English Nor	None
174	Descriptor	String	255	0		None	None

The procedure contains a deliberate mistake (which is included not so much as a warning, but to point out a strength of the syntax window). In work at City Hall, variables describing the characteristics of schools have been given the prefix 's' to distinguish them from variables describing the pupil characteristics, which have the prefix 'p'. Both variables in the syntax window have the prefix 's', and refer to schools. One of them should refer to pupil home area.

Fortunately, syntax in a Syntax window can be edited directly in the way that any text file can be edited, and Figure 101 shows that the variables have been changed to variables with a 'p' prefix. These are variables containing pupil home local authority information, and once that editing has been done, selecting 'Run' and 'All' in the Syntax window will set SPSS running frequencies tables for those variables.

Once a command has been run, the Syntax window will be minimised, and it can also be minimised directly by selecting the third button from the right at the top of the Syntax window. Put another way, and this is a key point, the Syntax window remains active until it is closed. This is particularly useful in the early days of work in Syntax. If further work is carried out in SPSS, any SPSS commands involved can be copied to an open Syntax window by selecting the 'Paste' button in the relevant procedure dialogue box as the procedure is carried out. Where a number of procedures need to be carried out more than once, you can build a single Syntax file which covers all the procedures, and which can be reused as needed.

Figure 102 shows, that a command for Frequency Tables run on pupil home area variables, and a command for a copy to be made (Compute) of the pupil home area code, have been added to the syntax window. The new commands can be run in the ordinary way, and the SPSS command syntax can be saved to a file with a name of the user's choice, and then re-opened and 'Run' at a later date. 'Running an SPSS syntax file on different datasets requires that datasets have the same set of data (even though, for example, for different points in time or parts of the country), and that the relevant variable names are the same. However, there is little point, if any, in using SPSS syntax for a purely one off exercise involving very few syntax commands.

There is another way which, while it is not free of risk, potentially reduces the number of copy and paste exercises. This involves changing SPSS options so that the SPSS commands you give during a working session are included in SPSS Output. Selecting 'Edit' from the main menu, followed by 'Options', opens the window shown in Figure 103. Select the 'Viewer' Tab, and then below 'Initial Output Status' and 'Contents are initially' select the 'Shown' radio button as shown in Figure 103.

Figure 101. Editing in the Syntax window

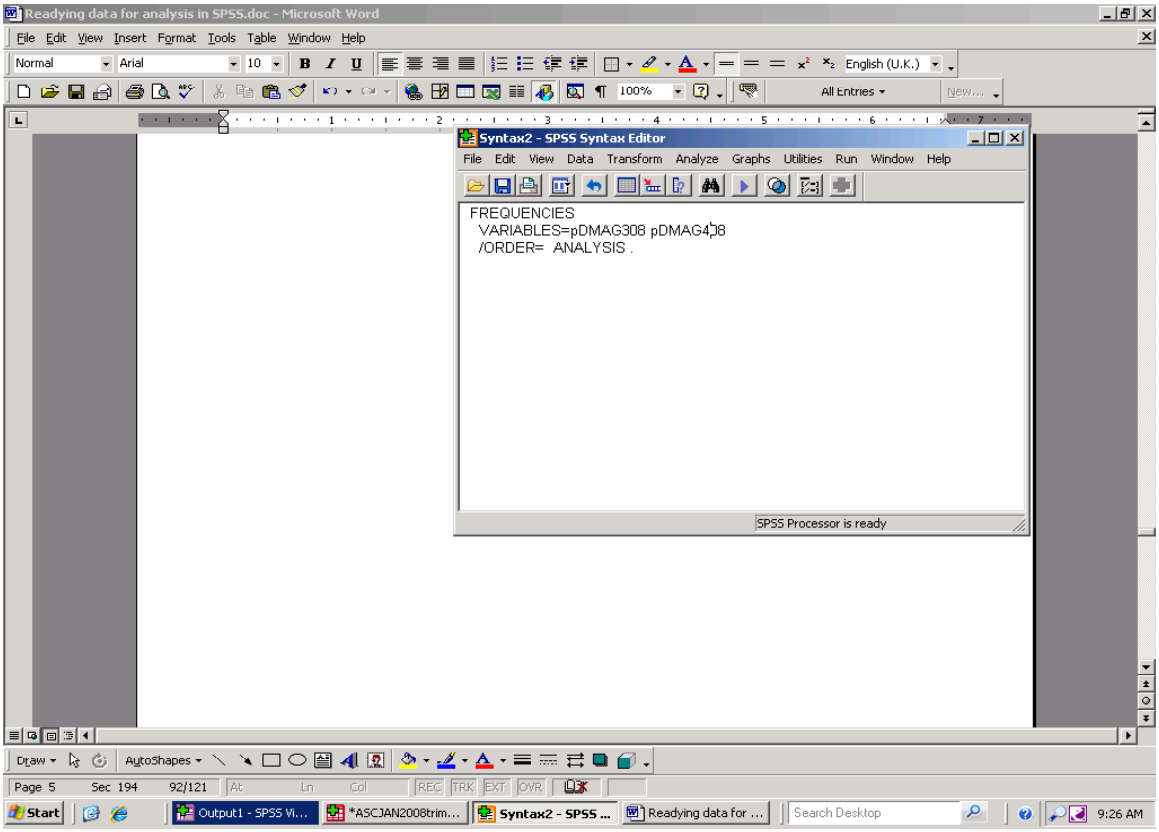
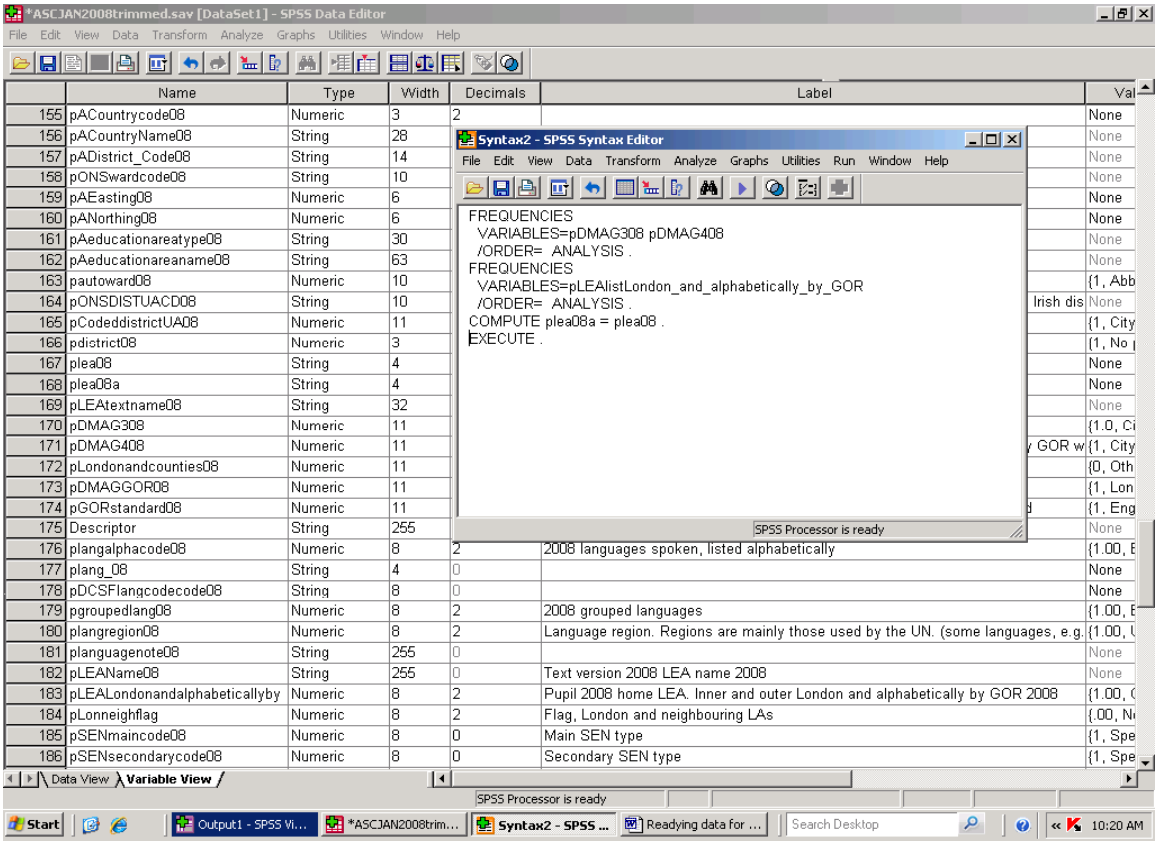


Figure 102. Pasting new syntax into an open Syntax window



Commands in the Output window can be copied to a word processing document for the record, or copied to a Syntax file. There is a good deal to be said for keeping a record in a word processing document if you are new to the type of work described in the Guide. Firstly, you will have a detailed record you can refer to. Secondly, it will not be held in a Syntax file, so there is no risk of it being accidentally activated with that that might entail for whatever dataset you might happen to have open. Lastly, you can copy the material back into a Syntax window if you wish.

Figure 103. Including SPSS commands in the Output file

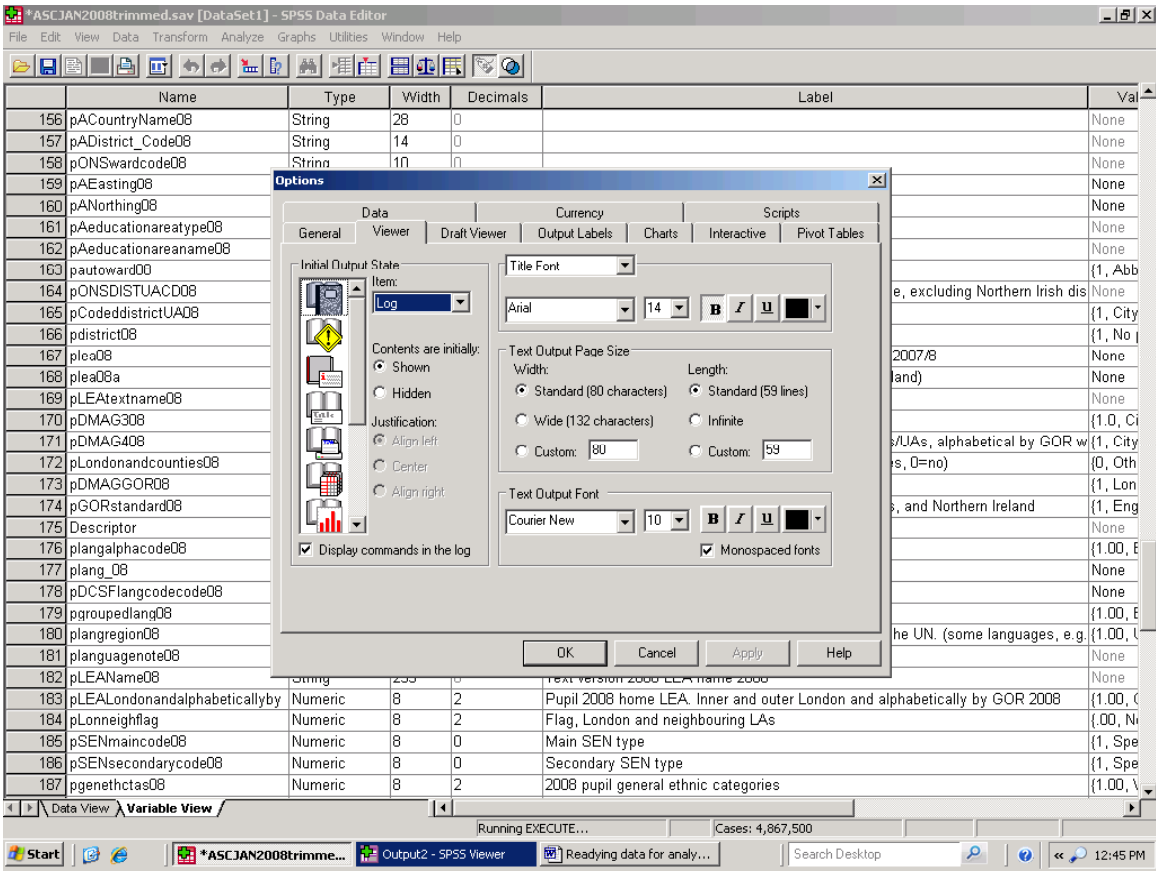


Figure 104 is an example of an SPSS Output file copied into Word to produce a clean record of a session of work. The Table, which would have formed the main part of the output originally, is now embedded in the list of the commands that led up to it, and that followed after it once it had been created. For those who want a fuller record, or who dislike even the reduced number of copy and paste steps involved, an alternative is to save output as a text file. When opened, the text file is more complex than that shown in Figure 104, but the alternative is there for those who wish to explore it. To copy command syntax from a word processing document to a wholly new Syntax window (or to key it in directly) select 'File' from the SPSS main menu, followed by 'New' and then 'Syntax'. A Syntax window will be shown, and work can proceed. Remember, Syntax windows are for SPSS commands and nothing else.

A further way of (a) creating a record of what has been done which can (b) in part be reused in a syntax file involves the SPSS journal file. This is a record, kept by SPSS automatically, of all the commands made in a working session. To locate that file select 'Edit' followed by 'Options' on the SPSS main menu. The location of the journal file is shown in the 'General' tab, in the 'Session journal' section. If you have access to that part of the computer, and you may well not, return to the SPSS main menu and Select 'File' followed by 'Open' followed by 'Other'. Selecting 'Other' gives access to all file types, and the journal file has the suffix 'jnl'. If you cannot locate the file, it may well mean that you do not have access to that part of the computer. If you can locate it, save it immediately under a new name, and close the original journal file. The *copied* version can then be edited as with any text file. Do not open and edit the file and then simply save it as the journal file. For those who wish to explore syntax in more detail, a reference guide can be opened selecting 'Help' in the main SPSS menu, followed by 'Command Syntax Reference'. The guide is more than 2,000 pages long, and whether it is available

will depend on whether it has been installed in the first instance.

Figure 104. SPSS Output including SPSS commands

```
IF (sla08=plea08a) pibob08 = 0 .
EXECUTE .
COMPUTE plea08a = plea08 .
EXECUTE .
IF (sla08=plea08a) ibob08 = 0 .
EXECUTE .
IF (sla08<>plea08a) ibob08 = 0 .
EXECUTE .
IF (sla08<>plea08a) ibob08 = 1 .
EXECUTE .
IF (pLEALondonandalphabeticallybyGOR=999) ibob08 = 2 .
EXECUTE .
SAVE OUTFILE='E:\SPSS\LPD EPD\National 2008 NPD\ASCJAN2008trimmed.sav'
 /COMPRESSED.
FREQUENCIES
  VARIABLES=pibob08
  /ORDER=  ANALYSIS .
RECODE
  pibob08  (MISSING=2)  .
EXECUTE .
FREQUENCIES
  VARIABLES=pibob08
  /ORDER=  ANALYSIS .
RECODE
  pibob08  (999=2)  .
EXECUTE .
FREQUENCIES
  VARIABLES=pibob08
  /ORDER=  ANALYSIS .

Frequencies

[DataSet1] E:\SPSS\LPD EPD\National 2008 NPD\ASCJAN2008trimmed.sav

                                Statistics

2008 pupil attends home LA school or non-home LA school
```

N	Valid	7512946
	Missing	0

2008 pupil attends home LA school or non-home LA school

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Attends school maintained by home LA	7020833	93.4	93.4	93.4
	Attends school maintained by another LA	440261	5.9	5.9	99.3
	No match on pupil home postcode	51852	.7	.7	100.0
	Total	7512946	100.0	100.0	

```
SAVE OUTFILE='E:\SPSS\LPD EPD\National 2008 NPD\ASCJAN2008trimmed.sav'
 /COMPRESSED.
```

Figure 105. Locating the SPSS journal file

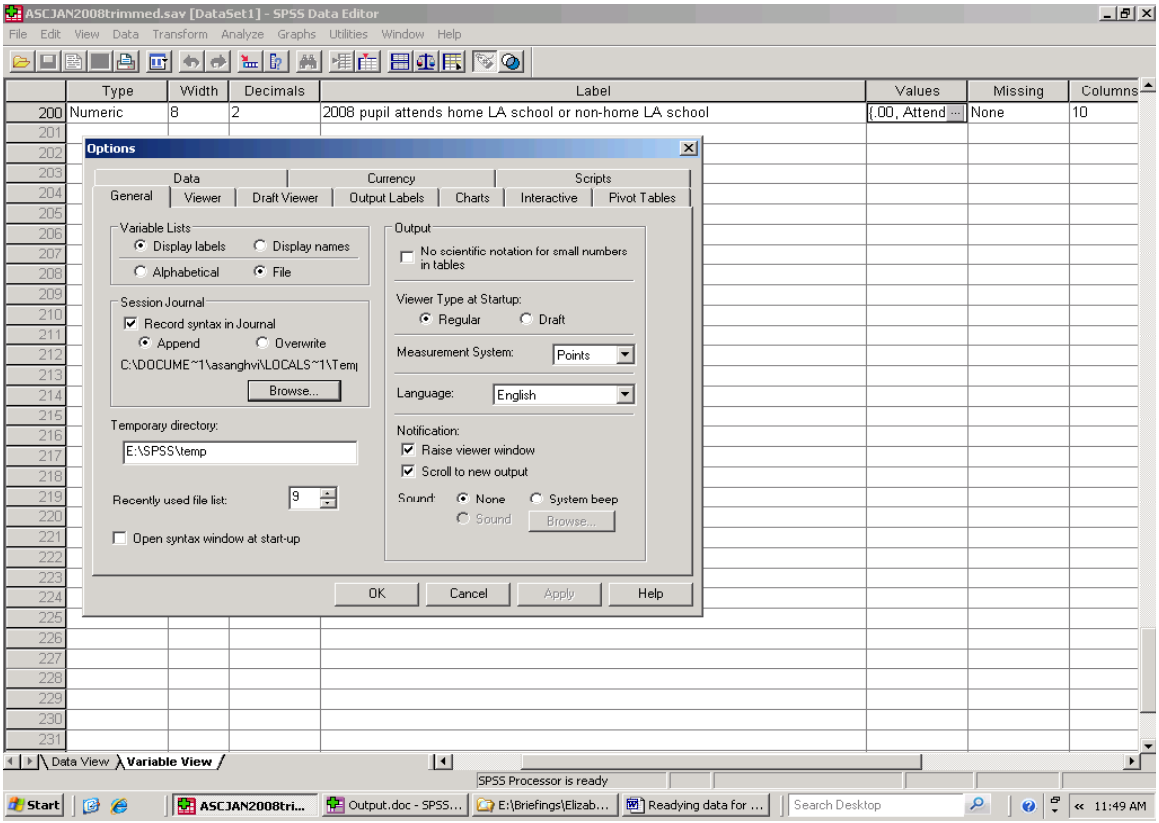
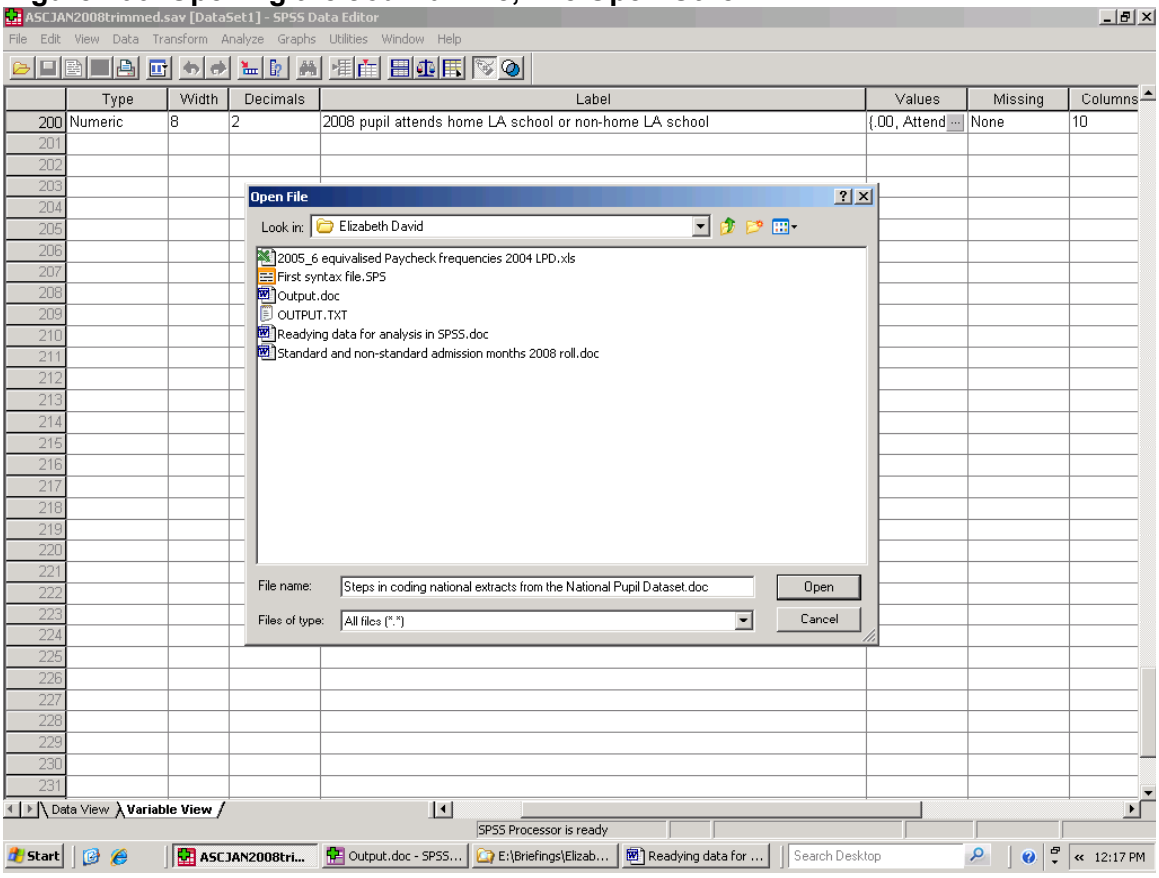


Figure 106. Opening the Journal File, File\Open\Other



Section 26. Conclusions and next steps

The Guide has stressed that readying data for analysis is itself located within a wider set of tasks and considerations. The extent to which individuals using the procedures introduced in the Guide will be involved in wider tasks, such as refining research questions, or negotiating access to data will vary. What will not vary is that those wider tasks and considerations will exist, and any researchers worth their salt will not want to ignore them.

Part of the reason why the SPSS procedures discussed can be so useful, is that some extremely important datasets exist, which are not necessarily in the form that a particular researcher might choose. Two types of example have been given, with the first being the situation where a dataset has been designed for purposes other than those the researcher has in mind. The second situation is where data are held in relational database software.

The National Pupil Dataset is held in a relational database warehouse, and some readers may have concluded from the Guide that relational databases are nothing but a problem for the researcher. The view taken here, may sound close to corporate rah! rah! but nonetheless the view is that relational databases present an opportunity and a challenge. Relational databases are more economic in their use of computing capacity than SPSS, but lack its functionality. They are widely used in government, in national services such as education and health, and in business. Relational databases are major repositories of information.

Anyone using data from relational databases should expect to cover at least some of the ground introduced in the Guide. For those who wish to know more about them, there is an abundance of written material on the web, with key issues being, in order of priority

- database design,
- programing structured query language (SQL),
- capacity, meaning how many records can any particular database software take
- data import and export capability
- and finally, database software.

Relational databases may appeal to those whose work and interests focuses on any or all of the following: producing descriptive table; who enjoy programming, and; or who work with extremely large datasets.

In terms of self help what the next step actually, is depends, unsurprisingly, on, as it were, where the reader is. The Guide has stressed the need for researchers to have an accurate and complete understanding of the variables they analyse, and point still applies. Where understanding is missing or incomplete, the first next step will be for the researcher to make good that shortfall good.

Regrettably, the pressure of work means that I will not be able to respond to requests for further advice about readying data for analysis. That said, the Guide has pointed to a number of steps the readers can take to help themselves.

The SPSS Help file can be accessed through the SPSS main menu. Sections 15 and 19 of the Guide point to the short explanatory notes that accompany the separate Functions in their Function Groups, and the SPSS Syntax Reference document, referred to in Section 25, can be accessed through ‘Help’ in the main SPSS menu, followed by ‘Command Syntax Reference’, if it has been installed on the machine being used. A further option, already available to the SPSS user, is to explore options that exist in SPSS dialogue boxes. Figure 91 on page 99 shows statistics options available with the SPSS Frequencies.

A range of other facilities in SPSS, which can be accessed via the SPSS main menu, also remain for the researcher to explore. These include its graphing capability, which are useful in, for example, identifying outliers (which takes us into the world of ‘if its interestingly different, its wrong’ type of view, and into the very different world of John Snow, the brewery as an outlier case, and the identification of cholera as a water borne disease). Other facilities in SPSS not discussed here, and accessible via the main SPSS menu, also provide statistics that are useful in refining research questions before full analysis begins. These include

- Explore (Analyse\Descriptives\Explore)
- Rank Cases (Transform\Rank Cases)
- OLAP Cubes (Analyse\Reports\OLAP Cubes) and
- Reports Analyse\Reports\Reports Summaries in Rows …

Section 1.1 noted that there is already a large number of books written about statistical analysis in SPSS. Where the reader needs to develop further statistical skills, advice may be available through a university, and a search of the internet on a phrase such as ‘Intermediate Statistical Analysis in SPSS’ or ‘Advanced Statistical Analysis in SPSS’ produces links to a number of books for sale and to other resources.

Jacqueline Collier's *Using SPSS Syntax: A beginner's Guide* will be published by Sage (London) in the near future (October 2009). It may well plug a literature gap and, more to the point, be useful for those who are just beginning work with SPSS syntax. At the next stage, Sarah E. Boslaugh has written *An Intermediate Guide to SPSS Programming: Using Syntax for Data Management* (Sage Publications, Thousand Oaks, California, 2005).

Raynald Levesque and associates at SPSS have written *Programming and Data Management for SPSS® Statistics 17.0: A Guide for SPSS Statistics and SAS® Users.* This is published by SPSS, and at the time of writing (August 2009) it is available as a free download from SPSS at http://www.spss.com/statistics/base/ProgDataMgmt_SPSS17.pdf

Raynald Levesque and associates at SPSS have also written a number of editions of *SPSS Programming and Data Management. A Guide for SPSS and SAS® Users,* also published by SPSS. The 2nd Edition was published in 2005, in response to the release of SPSS 13.0, and the 3rd edition was published in 2006 in response to the release of SPSS 14.0.1. The 4th edition was published in 2007 in response to the release of SPSS 15, and includes information on using SPSS command syntax in combination with the Python programming language. That combination requires SPSS 15.0.1 or later. Python is open source software, and information about the language is available at http://www/python.org/

For those with access to a university and using one or more of SPSS 13 to 15.0.1, that institution may already have the most relevant edition/s of *Programming and Data Management*, and it may also be possible to obtain the relevant edition over the Internet.

Appendix. Variable list, merged 2002 to 2005 London Pupil Dataset

Variable	Position	Label	Measurement Level
mergedid	1	Merged 2002 2003 spss id number	Scale
v2spssid	2	v2 $CASENUM 2002 PLASC	Scale
pl_ac	3	<none>	Nominal
pupcod2	4	copy of pupcode	Nominal
age02	5	Pupil age 2001/2002 school year	Scale
mth_age	6	<none>	Ordinal
age02514	7	Age in 2002 in 5-14 age range	Scale
age03615	8	Age in 2003 in 6-15 age range	Scale
pupgen	9	2002 Pupil Gender	Scale
pl_lea2	10	2002 school LEA	Scale
innerout	11	Inner London, Outer London or other school	Scale
pl_estab	12	<none>	Scale
pl_shid	13	Unique School id	Scale
seast02	14	school easting 2002	Scale
snorth02	15	school northing 2002	Scale
pl_join	16	<none>	Nominal
pl_post	17	<none>	Nominal
pl_ncyr	18	<none>	Nominal
dfesethc	19	Ethnic classification used in DfES publications	Scale
engeal2	20	Pupils first language is or is not English	Scale
fsm02	21	2002 FSM record	Scale
senstage	22	<none>	Ordinal
v2pflag	23	<none>	Scale
v2k1_ac	24	<none>	Nominal
v2age_yr	25	<none>	Ordinal
v2mthage	26	<none>	Ordinal
v2k1gend	27	<none>	Nominal
v2k1lea	28	<none>	Scale
v2k1estb	29	<none>	Scale
v2rtstp1	30	v2 k1 reading test point score (LDA-type)	Scale
v2wtstp1	31	v2 k1 writing test point score (LDA-type)	Scale
v2sptsp1	32	v2 k1 spelling test point score (LDA-type)	Scale
v2aep1	33	v2 k1 average English point score (LDA-type)	Scale
v2cep1	34	v2 k1 coded English test levels based on point scores (LDA)	Scale
v2cmp1	35	v2 k1 coded maths test (LDA-type)	Scale
v2k1e2p	36	v2 k1 English test 2 plus yes/no, point score based (as for LDA)	Scale
v2k1m2p	37	v2 k1 mathematics test 2 plus yes/no, point score based (as for LDA)	Scale
v2k1read	38	v2 k1 English reading task	Nominal
v2k1com	39	v2 k1 English comprehension test	Nominal
v2k1writ	40	v2 k1 English writing test	Nominal
v2k1spel	41	v2 k1 English spelling test	Nominal
v2k1math	42	v2 k1 maths test	Nominal
v2k1eta	43	v2 k1 English TA subject level	Nominal
v2k1ltta	44	v2 k1 English speaking and listening TA	Nominal
v2k1rta	45	v2 k1 English Reading TA	Nominal
v2k1wta	46	v2 k1 English writing TA	Nominal
v2k1mta	47	v2 k1 maths TA subject level	Nominal
v2k1mua	48	v2 k1 maths TA using and applying numbers	Nominal
v2k1ma	49	v2 k1 maths TA algebra	Nominal
v2k1msm	50	v2 k1 maths TA shapes and measures	Nominal
v2k1sta	51	v2 k1 science TA subject level	Nominal
v2k1se	52	v2 k1 science TA Experiment	Nominal
v2k1sclp	53	v2 k1 science TA Life processes	Nominal
v2k1smp	54	v2 k1 science TA material proporties	Nominal
v2k1spp	55	v2 k1 science TA physical processes	Nominal
v2k1flag	56	V2 k1 flag	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
v2k2_ac	57	<none>	Nominal
v2k2yage	58	<none>	Ordinal
v2k2mage	59	<none>	Ordinal
v2k2gend	60	<none>	Nominal
v2k2_lea	61	<none>	Scale
vk2estab	62	<none>	Scale
v2k2ma	63	<none>	Nominal
v2k2sc	64	<none>	Nominal
v2k2enta	65	v2 k2 English TA level	Nominal
k2mat_ta	66	v2 k2 maths TA level	Nominal
k2sci_ta	67	v2 k2 science TA level	Nominal
k2wel_ta	68	<none>	Ordinal
k2lev_e	69	v2 k2 English final test level	Nominal
k2lev_m	70	v2 k2 maths final test level	Nominal
k2lev_s	71	v2 k2 science final test level	Nominal
v2k2ect	72	v2 k2 coded English final test level	Scale
v2k2mct	73	v2 k2 coded maths final test level	Scale
v2k2sct	74	v2 k2 coded science final test level	Scale
v2k2e4p	75	Is there a v2 k2 English test at level 4p yes/no	Scale
v2k2m4p	76	Is there a V2 k2 mathematics test at level 4p yes/no	Scale
v2k2s4p	77	Is there a V2 k2 science test at level 4p yes/no	Scale
vk2lwe	78	<none>	Ordinal
vk2erl	79	v2 k2 English reading test level	Nominal
v2k2lewl	80	v2 k2 English writing test level	Nominal
v2ertm	81	v2 k2 English reading test mark	Nominal
v2k2ewtm	82	v2 k2 English writing test mark	Nominal
v2k2ehtm	83	v2 k2 English hand-writing test mark	Nominal
v2k2estm	84	v2 k2 English spelling test mark	Nominal
v2k2etm	85	v2 k2 English total mark	Nominal
v2k2et	86	v2 k2 English tier	Nominal
v2k2emtl	87	v2 k2 English main test level	Nominal
v2k2eetm	88	v2 k2 English extension test mark	Scale
v2k2eetl	89	v2 k2 English extension test level	Ordinal
v2k2mtma	90	v2 k2 maths test A mark	Nominal
v2k2mtmb	91	v2 k2 maths test B mark	Nominal
v2k2mmam	92	v2 k2 maths mental arithmetic mark	Nominal
v2k2mtm	93	v2 k2 maths total mark	Nominal
v2k2mt	94	v2 k2 maths tier	Nominal
v2k2mmtl	95	v2 k2 maths main test level	Nominal
v2k2metm	96	v2 k2 maths extenstion test mark	Scale
v2k2metl	97	v2 k2 maths extension test level	Ordinal
v2k2stma	98	v2 k2 science test A mark	Scale
v2k2stmb	99	v2 k2 science test B mark	Nominal
v2k2stm	100	v2 k2 science total mark	Nominal
v2k2st	101	v2 k2 science tier	Nominal
v2k2smtl	102	v2 k2 science main test level	Nominal
v2k2setm	103	v2 k2 science extension test mark	Nominal
v2k2setl	104	v2 k2 science extension test level	Ordinal
v2k2flag	105	k2 v2 flag	Scale
v2k3_ac	106	v2 k3 year assessed	Nominal
v2k3agey	107	v2 k3 age_s_yr	Ordinal
v2k3agem	108	v2 k3 mth_age	Ordinal
v2k3gend	109	v2 k3_gend	Nominal
v2k3_lea	110	v2 k3_lea	Scale
v2k3est	111	v2 k3_estab	Scale
v2k3schl	112	v2 k3_schid	Scale
v2k3eta	113	v2 k3 English TA	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
v2k3mta	114	v2 k3 maths TA	Nominal
v2k3sa	115	v2 k3 sci TA	Nominal
k3wel_ta	116	k3wel_ta	Nominal
v2k3etl	117	v2 k3 English final test level	Nominal
v2k3mtl	118	v2 k3 maths final test level	Nominal
v2kestl	119	v2 k3 science final test level	Nominal
v2k3ect	120	v2 k3 coded English final test levels	Scale
v2k3mct	121	v2 k3 coded maths final test levels	Scale
v2k3sct	122	v2 k3 coded science final test levels	Scale
v2k3e5p	123	Is there a v2 k3 English test at level 5p yes/no	Scale
v2k3m5p	124	Is there a v2 k3 mathematics test at level 5p yes/no	Scale
v2k3s5p	125	Is there a v2 k3 science test at level 5p yes/no	Scale
k3lev_we	126	v2 k3lev_we	Nominal
v2k2p1m	127	v2 k3 English paper 1 mark	Nominal
v2k3ep2m	128	v2 k3 English paper 2 mark	Nominal
v2k3etm	129	v2 k3 English total mark	Nominal
v2k3et	130	v2 k3 English tier	Nominal
v2k3emtl	131	v2 k3 English main test level	Nominal
v2k3extm	132	v2 k3 English extension test mark	Nominal
v2k3extl	133	v2 k3 English extension test level	Nominal
v2k3mp1m	134	v2 k3 maths paper 1 mark	Nominal
v2k3mp2m	135	v2 k3 maths paper 2 mark	Nominal
v2k3mam	136	v2 k3 maths mental arithmetic mark	Nominal
v2k3mtm	137	v2 k3 maths total mark	Nominal
v2k3mt	138	v2 k3 maths tier	Nominal
v2k3mmtl	139	v2 k3 maths main test level	Nominal
v2k3mxtm	140	v2 k3 maths extension test mark	Nominal
v2k3mxtl	141	v2 k3 maths extension test level	Nominal
v2k3stam	142	v2 k3 science test A mark	Nominal
v2k3stbm	143	v2 k3 science test B mark	Nominal
v2k3stm	144	v2 k3 science total mark	Nominal
v2k3st	145	v2 k3 science tier	Nominal
v2k3smtl	146	v2 k3 science main test level	Nominal
v2k3sxtm	147	v2 k3 science extension test mark	Nominal
v2k3sxtl	148	v2 k3 science extension test level	Nominal
v2k3flag	149	v2 k3 flag	Scale
v2pupcod	150	v2 PLASC id	Nominal
schdmag1	151	v2 School LEA DMAG1	Scale
schdmag2	152	v2 School LEADMAG2	Scale
schgor	153	v2 School Government Office for the Regions	Scale
k4c_ac	154	<none>	Nominal
k4c_sex	155		Nominal
k4gen	156	<none>	Scale
k4_yrgrp	157	<none>	Ordinal
k4c_pem	158	<none>	Nominal
lat_cen	159	<none>	Nominal
lat_dfes	160	<none>	Scale
k4_schid	161	<none>	Scale
k4i_lea	162	<none>	Scale
k4_estab	163	<none>	Scale
k4_gcse_	164	<none>	Ordinal
gcse_sho	165	<none>	Ordinal
gnvq_ful	166	<none>	Ordinal
gnvq_f_f	167	<none>	Ordinal
gnvq_p1	168	<none>	Ordinal
gnvq_lan	169	<none>	Ordinal
gvq_lani	170	<none>	Ordinal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
nc_cert	171	<none>	Ordinal
pointss	172	<none>	Scale
pointsn	173	<none>	Ordinal
pointsc	174	<none>	Scale
gcse_pas	175	<none>	Ordinal
gcse_pa	176	<none>	Ordinal
gcse_pb	177	<none>	Ordinal
gcse_pc	178	<none>	Ordinal
gcse_pd	179	<none>	Ordinal
gcse_pe	180	<none>	Ordinal
gcse_pf	181	<none>	Ordinal
gcse_g	182	<none>	Ordinal
gnvqastc	183	<none>	Ordinal
gnvqdstc	184	<none>	Ordinal
high_eng	185	<none>	Ordinal
high_mat	186	<none>	Ordinal
totastc	187	Number of GCSE or equiv A* to C grades	Ordinal
allastc	188	<none>	Ordinal
astc5	189	<none>	Nominal
totastg	190	Number of GCSE or equiv A* to G grades	Ordinal
allastg	191	<none>	Ordinal
astg5	192	<none>	Nominal
gnvqastg	193	<none>	Ordinal
astgcp5	194	<none>	Nominal
astgem5	195	<none>	Nominal
tot_gcse	196	<none>	Scale
mea_gcse	197	<none>	Scale
gnvq_eaa	198	<none>	Ordinal
gnvq_eb	199	<none>	Ordinal
gnvq_ec	200	<none>	Ordinal
gnvq_ed	201	<none>	Ordinal
gnvq_ee	202	<none>	Ordinal
gnvq_efg	203	<none>	Ordinal
k4flag	204	k4flag	Scale
nastc	205	New 5 A*-C measure - missing values equal 0	Scale
npoint	206	New uncapped point score - missing value equals 0	Scale
grppoint	207	Grouped point score ranges	Scale
bandpnt	208	Banded point score range	Scale
npointc	209	New capped point score - missing values equal 0	Scale
ntotag	210	New 1+ A*-G - missing values equal 0	Scale
atog5p	211	Achieved 5+ A*-G or equivalent	Nominal
ppcode02	212	Pupil home postcode 2002	Nominal
peast2	213	2002 pupil home easting	Scale
pnorth2	214	2002 pupil home northing	Scale
lea	215	2002 pupil home LEA spss code	Scale
spssid	216	<none>	Scale
onswdcd	217	pupil home 2002 ONS ward code	Nominal
ward_nam	218	pupil home 2002 ward name	Nominal
onsdcd	219	pupil home ONS ward code	Nominal
district	220	pupil home 2002 district	Nominal
spss02wd	221	pupil coded home 2002 wards	Scale
spsswdis	222	pupil coded home 2002 London wards, with other districts/UAs	Scale
govtype	223	pupil home 2002 government type	Ordinal
englea	224	pupil home LEA	Scale
leadmag1	225	pupil grouped home LEA (LEADMAG1)	Scale
leadmag2	226	pupil grouped home LEA (LEADMAG2)	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
gor	227	pupil home Government Office for the Regions	Scale
schlflag	228	Assessment records with & without PLASC data	Scale
edschl	229	EduBase school name	Nominal
town	230	EduBase school town	Nominal
pcode	231	EduBase school postcode	Nominal
open_clo	232	School open or closed	Ordinal
whyopen	233	New school - why opened?	Scale
whyclose	234	School closed - why?	Scale
schopend	235	School opening date	Scale
schclose	236	School closing date	Scale
toe_code	237	School type (Community, VA ...)	Scale
poe_code	238	School phase	Ordinal
phase	239	<none>	Scale
special	240	mainstream or special school	Scale
low_age	241	School age range - low age	Ordinal
high_age	242	School age range - high age	Ordinal
gender	243	Intake gender	Ordinal
total_pu	244	Total pupils	Scale
total_gi	245	Total girls	Scale
total_bo	246	Total boys	Scale
denom	247	Denomination	Ordinal
app_spec	248	APP_SPEC_PUPILS	Scale
adpolicy	249	Admissions policy	Ordinal
spec_cla	250	Special classes?	Ordinal
school_c	251	School capacity	Scale
lsc_area	252	EduBase Learning and Skills Council Area	Scale
spclism	253	Specialist school	Ordinal
specmeas	254	School on special measures	Ordinal
eaz	255	School Education Action Zone	Scale
beacon	256	Beacon school	Ordinal
fresh_st	257	Fresh start school	Ordinal
id	258	ID	Scale
srawpost	259	School raw postcode	Nominal
swardcod	260	School ward code by location of school	Nominal
sdistric	261	school district by location of school	Nominal
swardnam	262	school's ward name by location of code	Nominal
sgovtype	263	school gov region by location of school	Ordinal
sonsdist	264	school's ONS district code by location of school	Nominal
senglea	265	School English LEA label by location of school	Scale
snatauth	266	School nat authority label by location of school	Scale
dmag1	267	School area code DMAG by location of school	Scale
sleagor	268	School GOR by location of school	Nominal
plaschl	269	<none>	Scale
elea_1	270	ENGLEA	Scale
govtyp_1	271	2002 gov type	Ordinal
englea_1	272	2002 English LEA	Scale
leadma_1	273	2002 LEA DMAG1	Scale
leadma_2	274	2002 LEA DMAG2	Scale
gor_1	275	2002 Government Office for the Regions	Scale
number	276	<none>	Scale
estab	277	ESTAB	Scale
schoolna	278	SCHOOLNAME	Nominal
pl_schid	279	School unique id	Scale
scl02wd	280	school spss code 2002 wards	Scale
sclwddis	281	school spss code 2002 London wards and other districts	Scale
schlgov	282	school 2002 gov type	Ordinal
sclea1	283	school 2002 English LEA	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
scdmag1	284	school 2002 LEA DMAG1	Scale
scdmag2	285	school 2002 LEA DMAG2	Scale
scgor	286	2002 Government Office for the Regions	Scale
spost02	287	School edited postcode	Nominal
urn_a	288	URN_A	Scale
pupschl	289	name of school attended	Nominal
constitu	290	School attended, in which constituency	Nominal
whenbeac	291	Date school attended become a beacon school	Scale
beacflag	292	Flag for Sept 2002 Beacon School	Scale
ibob02	293	2002 Pupil at in-borough or out-borough school	Scale
flag02	294	2002 record flag	Scale
flag03	295	2003 record flag	Scale
spssid3	296	SPSS id code (on source file)	Scale
ac_03	297	PLASC school year	Nominal
pupcode	298	2004 pupil alphanumeric code	Nominal
pupcode3	299	<none>	Nominal
agejun02	300	<none>	Scale
tmage03	301	<none>	Scale
age03	302	Pupil age 2002/2003 school year	Scale
age3515	303	2003 pupils in 5-15 age range	Scale
month_03	304	age - months in addition to years, 31st August 2002	Nominal
gender03	305	<none>	Scale
gend_03	306	gender	Nominal
sla03	307	2003 LEA of school attended	Scale
sla03a	308	2003 actual LEA of school attended	Scale
sdamg13	309	School actual LEA/LEA group (1)	Scale
sdmag23	310	Grouped LEA codes	Scale
sestab03	311	School code	Scale
schid_03	312	Combined LEA and school code	Scale
seast03	313	school 2003 easting	Scale
snorth03	314	school 2003 northing	Scale
join_03	315	Pupil admission date	Nominal
dd03	316	Day of week admitted	Nominal
mm03	317	Month of year admitted	Nominal
yy03	318	Year amitted	Nominal
admit	319	Admission date	Scale
ppcode03	320	Edited pupil home postcode	Nominal
peast03	321	pupil 2003 home easting	Scale
pnorth03	322	pupil 2003 home northing	Scale
pti_03	323	Pupil part-time indicator	Nominal
dfeseth3	324	DfES ethnicity source code	Nominal
engeal3	325	Pupil's first language	Scale
fsm03	326	2003 FSM record	Scale
sen_03	327	Pupil SEN stage	Nominal
enrol_03	328	Pupil enrolment status	Nominal
ncyr_03	329	Pupil national curriculum year group	Nominal
conas_03	330	Connexions assent	Nominal
pa_03	331	Pupil post-A level indicator	Nominal
alev_03	332	Pupil - N. A levels being studied	Nominal
gcse_03	333	Pupil - N. GCSEs being studied	Nominal
gnvq_03	334	Taking GNVQ by level	Nominal
pgnvq_03	335	Taking GNVQ precurso	Nominal
nvq_03	336	Taking NVQ by level	Nominal
oth_03	337	Other post 16 course being taken	Nominal
lea_res3	338	Pupil home LEA (?)	Scale
count3	339	<none>	Scale
main3	340	<none>	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
spssme3	341	DfES main ethnic groups	Scale
spssse3	342	DfES ethnic subcategories	Scale
spssx3	343	DfES extended ethnic codes with main categories	Scale
spssx13	344	DfES extended ethnic codes with subcategories	Scale
divis3	345	ethnic category is divisible	Scale
divisoth	346	ethnic category is divisible or an 'any other' type	Scale
oldeth3	347	Old ethnic category (reserved for excluded pupils)	Scale
ponswcd3	348	pupil home ONS ward code	Nominal
ponsdis3	349	Pupil home ONS district code	Nominal
pgovtyp3	350	Pupil home government type	Ordinal
plea03	351	Pupil home LEA - postcode-based	Scale
pdmag103	352	2003 Pupil grouped grouped LEA (1) - postcode-based	Scale
pleadg23	353	2003 Pupil grouped home LEA (2) - postcode-based	Scale
pinoutoth03	354	Pupil home area - inner London, outer London, other	Scale
pgor3	355	Pupil home GOR - postcode-based	Scale
psp02w3	356	Coded pupil home 2003 ward - postcode-based	Scale
pspwds3	357	Coded pupil home 2003 London wards and other districts - postcode-based	Scale
pwname3	358	Pupil home ward (text) - postcode based	Nominal
wardlrc3	359	Pupil pre-2002 ward code	Nominal
fsmcod13	360	<none>	Scale
spcase3	361	Spss 2003 case number	Scale
urn3	362	URN	Scale
pinoutother03	363	LEA type. Inner London, Outer London or other	Scale
london3	364	London LEA	Scale
schlea3	365	LEA	Scale
schlna3	366	2003 school	Nominal
Spcode3	367	EduBase school postcode	Nominal
spost3	368	Working postcode	Nominal
openclo3	369	School open or closed	Ordinal
schopen3	370	School opening date	Scale
schclos3	371	School closing date	Scale
stoe03	372	School type (Community, VA ...) 2003	Scale
sadauth03	373	2003 pupil admissions authority	Scale
spoecode3	374	School phase	Ordinal
sphase3	375	primary or secondary	Scale
smorind3	376	maintained or independent school	Scale
lowage3	377	School age range - low age	Ordinal
shighage3	378	School age range - high age	Ordinal
sgender3	379	Intake gender	Ordinal
stotalpu3	380	Total pupils	Scale
stotalgi3	381	Total girls	Scale
stotalbo3	382	Total boys	Scale
sdenom3	383	Denomination	Ordinal
snurspro3	384	Nursery provision	Ordinal
schoolc3	385	School capacity	Scale
sedbward3	386	School EDB school ward text code	Nominal
swarddes3	387	School EDB school ward	Nominal
sedbdist3	388	School EDB school district code	Nominal
sdistri13	389	School EDB school district	Nominal
slscarea3	390	School EDB Learning and Skills Council Area	Scale
slsc_ar13	391	School EDB LSC_AREA_DESC	Nominal
splstcd3	392	2003 Specialist school type	Scale
specmea3	393	Schools on special measure	Ordinal
eaz3	394	School Education Action Zone	Scale
spflag3	395	postcode extract flag	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
senglea3	396	School LEA - postcode-based	Scale
sldmag1p	397	School LEADMAG1 - postcode-based	Scale
sldmag2p	398	School LEADMAG2 - postcode-based	Scale
s02wdcp3	399	School spss code 2003 wards - postcode-based	Scale
s02wddp3	400	School spss code 2003 London wards and other districts - postcode-based	Scale
sonswd3	401	School ONS ward code	Nominal
sonswdp3	402	<none>	Nominal
sonsdsp3	403	<none>	Nominal
sdistp3	404	School district postcode-based	Nominal
sgovtyp3	405	School GOVTYPE postcode-based	Ordinal
swdlrcp3	406	School 1996 London ward postcode-based	Nominal
londonfl	407	<none>	Scale
npcdmis3	408	No postcode match (2003 LPD)	Scale
pupmatch	409	2002 and 2003 pupil codes match or do not match	Scale
match123	410	2002 and 2003 records not including unattached GCSE pupil records	Scale
miss2002	411	Pupil in merged dataset with/without 2002 record	Scale
miss2003	412	Pupil in merged dataset with/without 2003 record	Scale
pmatch	413	Merged file - pupil with or without same postcode in 2002 and 2003	Scale
leamatch	414	Merged file - pupil's school in same LEA 2002 and 2003	Scale
sclmatch	415	Merged file - pupil's school same in 2002 and 2003	Scale
phome23	416	Stability and mobility, across & within L.A. areas	Scale
schid02	417	2002 unique school id	Scale
seast2	418	2002 school six digit easting	Scale
snorth2	419	2002 school six digit northing	Scale
pcode2	420	Pupil home postcode	Nominal
spost2	421	School edited postcode	Nominal
schid03	422	2003 unique school id	Scale
seast3	423	2003 school six digit easting	Scale
snorth3	424	2003 school six digit northing	Scale
peast3	425	pupil 2003 home easting	Scale
pnorth3	426	pupil 2003 home northing	Scale
east2sq	427	<none>	Scale
north2sq	428	<none>	Scale
hmschl2	429	2002 distance (metres) between pupil home and school	Scale
east3sq	430	<none>	Scale
north3sq	431	<none>	Scale
hmschl3	432	2003 distance (metres) between pupil home and school	Scale
hh23esq	433	<none>	Scale
hh23nsq	434	<none>	Scale
hh23	435	Distance (metres) between 2002 and 2003 pupil home. Only pupils with full co-ordinates.	Scale
hs23esq	436	<none>	Scale
hs23nsq	437	<none>	Scale
hs23	438	Distance (metres) between pupil 2002 home and 2003 school	Scale
ss23esq	439	<none>	Scale
ss23nsq	440	<none>	Scale
ss23	441	Distance (meters) between pupil 2002 and 2003 school	Scale
dist23	442	distance record complete 2002 and 2003	Scale
dist2002	443	distance record for 2002 complete	Scale
dist2003	444	distance record for 2003 complete	Scale
problem	445	<none>	Scale
casad3	446	<none>	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
p2002io	447	Pupil 2002 home in inner or outer London	Scale
p2003io	448	Pupil 2003 home in inner or outer London	Scale
s2002io	449	2002 school in inner or outer London	Scale
s2003io	450	2003 school in inner or outer London	Scale
schl2002	451	2002 school	Nominal
schl2003	452	2003 school	Nominal
gtotal	453	Grand total (count)	Scale
pban12	454	<none>	Scale
ppcode2	455	Pupil home postcode	Nominal
wrdcd	456	WRDCD	Nominal
wrd	457	WRD	Nominal
distoa	458	District - OA-based	Nominal
wardoa	459	Ward OA-based	Nominal
wdcd	460	WDCD	Nominal
oa_code	461	oa-code	Nominal
pcprofma	462	HRP 26-64 % professional or managerial (excludes unclassifieds)	Scale
pcint	463	HRP 26-64 % intermediate, small employers, lower supervisory (excludes unclassifieds)	Scale
pcman	464	HRP 26-64 % semi-routine, routine, never worked/long-term unemployed (excludes unclassifieds)	Scale
prof50p	465	HRP 26-64 OA >50% professionals/managers (excludes unclassifieds)	Scale
int50p	466	HRP 26-64 OA >50% intermediate, small employers, lower supervisory (excludes unclassifieds)	Scale
man50p	467	HRP 26-64 OA >50% semi-routine, routine, never worked/long-term unemployed (excludes unclassifieds)	Scale
prof60p	468	HRP 26-64 OA >60% professionals/managers (excludes unclassifieds)	Scale
int60p	469	HRP 26-64 OA >60% intermediate, small employers, lower supervisory (excludes unclassifieds)	Scale
man60p	470	HRP 26-64 OA >60% semi-routine, routine, never worked/long-term unemployed (excludes unclassifieds)	Scale
profrnge	471	HRP 26-64 professional/managerial grouped by % in OA (excludes unclassifieds)	Scale
intrnge	472	HRP 26-64 intermediate, small employers, lower supervisory, grouped by % in OA (excludes unclassifieds)	Scale
manrnge	473	HRP 26-64 semi-routine, routine, long-term unemployed, grouped by % in OA (excludes unclassifieds)	Scale
mobtest1	474	No Harrow. Pupils with 2002 age 4-5, 7-9, 11-14 or 2003 age 5-6, 8-10, 12-15	Scale
mobtest2	475	Dom schl mobility type	Scale
age2313	476	Pupil aged 13 in 2002 or 14 in 2003	Scale
temper	477	<none>	Scale
pexclude03	478	2003 permanent exclusion	Nominal
enrol03	479	2003 enrollment status	Nominal
ncyr03	480	2003 national curriculum year group	Nominal
care03	481	2003 pupil in care flag	Nominal
schlcare03	482	2003 pupil in care at current school flag	Nominal
LEAcare03	483	2003 care authority	Nominal
ks103flag	484	ks1 2003 flag	Scale
k103spsscode	485	k1 2003 spss code (same order as alphnumeric code)	Scale
k1_pmr03	486	k1 2003 DfES alphanumeric code	Nominal
k1_ac03	487	<none>	Nominal
age_s_yr03	488	<none>	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
mth_ag03	489	<none>	Nominal
k1_gend03	490	<none>	Nominal
k1_lea03	491	<none>	Scale
k1_estab03	492	<none>	Scale
k1_schid03	493	<none>	Scale
k1read2p03	494	ks1 2003 reading task, pupil above/below level 2	Scale
k1comp2p03	495	ks1 2003 English Comprehension test, pupil above/below level 2	Scale
k1writ2p03	496	ks1 2003 English writing test, pupil above/below level 2	Scale
k1maths2p03	497	ks1 2003 maths test/task, pupil above/below level 2	Scale
k1engTA2p03	498	ks1 2003 English TA, pupil above/below level 2	Scale
k1mathsTA2p03	499	ks1 2003 maths TA, pupil above/below level 2	Scale
k1sciTA2p03	500	ks1 2003 science TA, pupil above/below level 2	Scale
ks203flag	501	ks2 2003 dataset flag	Scale
k203spsscode	502	ks2 2003 spps code, same order as alphanumeric code	Scale
k2_pmr	503	ks 2003 DfES alphanumeric code	Nominal
k2ac03	504	<none>	Nominal
ageyrs03	505	<none>	Nominal
mthage03	506	<none>	Nominal
k2gend03	507	<none>	Nominal
k2lea03	508	<none>	Scale
k2estab03	509	<none>	Scale
k2schid03	510	<none>	Scale
tot2e03	511	ks2 pupil 2003 total English mark	Nominal
tot2m03	512	ks2 pupil 2003 total maths mark	Nominal
tot2s03	513	ks2 pupil 2003 total science mark	Nominal
k2engTA2p03	514	ks2 2003 English TA, above or below level 4	Scale
k2mathsTA2p03	515	ks2 2003 Maths TA, above or below level 4	Scale
k2sciTA2p03	516	ks2 2003 science TA, above or below level 4	Scale
k2engtest2p03	517	ks2 pupil 2003 final English test level, above or below level 4	Scale
ks2mathstest2p03	518	ks2 pupil 2003 final maths test level, above or below level 4	Scale
k2scitest2p03	519	ks2 pupil 2003 final science test level, above or below level 4	Scale
tier_2e03	520	ks2 english 'tier'	Nominal
tier_2m03	521	ks2 maths 'tier'	Nominal
tier_2s03	522	ks2 science 'tier'	Nominal
k303flag	523	ks3 2003 dataset flag	Scale
k303spsscode	524	ks3 2003 pupil spss code - same order as DfES alphanumeric code	Scale
k3pmr03	525	<none>	Nominal
K3ac03	526	<none>	Nominal
k3ageyrs03	527	<none>	Nominal
k3mthage03	528	<none>	Nominal
K3gend03	529	<none>	Nominal
K3lea03	530	<none>	Scale
K3Estab03	531	<none>	Scale
K3schid03	532	<none>	Scale
k3engta03	533	ks3 2003 pupil English TA level	Nominal
K3matta03	534	ks2 2003 pupil maths TA level	Nominal
K3scita03	535	ks3 2003 pupil science TA level	Nominal
k3leve03	536	ks3 2003 pupil final English test level	Nominal
K3levm03	537	ks2 2003 pupil final maths test level	Nominal
K3levs03	538	ks3 2003 pupil final science test level	Nominal
k3tote03	539	ks3 2003 pupil total English mark - NV = Null Value	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
k3totm03	540	ks3 2003 pupil total maths mark - NV = Null Value	Nominal
k3tots03	541	ks3 2003 pupil total science mark - NV = Null Value	Nominal
k3engmark03	542	K3 English mark 2003 - no record = 0	Nominal
k3mathsmark03	543	K3 maths mark 2003 - no record = 0	Nominal
k3scimark03	544	K3 science mark 2003 - no record = 0	Nominal
k3avpoints03	545	k3 average points 2003 (subject total marks/3)	Scale
k3equarts03	546	k3 English mark quartiles 2003	Scale
k3mquarts03	547	k3 maths mark quartiles 2003	Scale
k3squarts03	548	k3 science mark quartiles 2003	Scale
k3pointquarts03	549	k3 average point score quartiles 2003	Scale
k3Engtier03	550	k3 2003 pupil English tier	Scale
k3mathstier03	551	k3 2003 pupil maths tier	Scale
k3sciencetier03	552	k3 2003 pupil science tier	Nominal
k3engTA5p03	553	k3 2003 pupil English TA, below or at/above level level 5	Scale
k3mathTA5p03	554	k3 2003 pupil maths TA, below or at/above level level 5	Scale
k3sciTA5p03	555	k3 2003 pupil science TA, below or at/above level level 5	Scale
k3engtest5p03	556	k3 2003 pupil final English test level, below or at/above level 5	Scale
k3mathstest5p03	557	k3 2003 pupil final maths test level, below or at/above level 5	Scale
k3scitest5p03	558	k3 2003 pupil final science test level, below or at/above level 5	Scale
k403flag	559	ks4 2003 dataset flag	Scale
k403spsscode	560	ks4 2003 spss code - same order as 2003 pupil alphanumeric code	Scale
k4pmr03	561	<none>	Nominal
k4cac03	562	<none>	Nominal
k4yrgroup03	563	<none>	Nominal
k4latcen03	564	Latest centre	Nominal
k4latdfes03	565	Latests DfES	Scale
k4land03	566	ks4 country (all records are for England)	Nominal
k4ilea03	567	2003 ks4 candidate raw LA	Scale
k4estab03	568	2003 ks4 candidate raw school	Scale
k4sex03	569	2003 ks4 candidate gender	Scale
k4schid03	570	2003 ks4 candidate raw LA and school identifier	Scale
k45acems03	571	2003 ks4 candidate with "5+ A*-C" including passes at grades A*-C in English, maths and science	Scale
k4fiveac03	572	2003 Ks4 pupil achieved five or more A*-C passes	Nominal
k4fiveag03	573	2003 ks4 pupil achieved five or more passes at A*-G	Nominal
k4ptstoldc03	574	2003 ks4 pupil old point scores, capped 8 best GCSE and GNVQ results	Nominal
k4entfgcse03	575	2003 ks4 number of pupil entries - full gcses	Nominal
k4enthgcse03	576	2003 ks4 number of pupil entries - half gcses	Nominal
k4entfintGNVQ03	577	2003 ks4 number of pupil entries - Full intermediate GNVQ	Nominal
k4entffoundationGNVQ	578	2003 ks4 number of pupil entries - Full foundation GNVQ	Nominal
k4entvpi03	579	2003 ks4 number of pupil entries - part intermediate GNVQ	Nominal
k4entvpf03	580	2003 ks4 number of pupil entries - Part 1 foundation GNVQ	Nominal
k4gcseastar03	581	2003 ks4 number of pupil GCSE grade A* passes	Nominal
k4gcsea03	582	2003 ks4 number of pupil GCSE grade A passes	Nominal
k4gcseb03	583	2003 ks4 number of pupil GCSE grade B passes	Nominal
k4gcsec03	584	2003 ks4 number of pupil GCSE grade C passes	Nominal
k4gcsed03	585	2003 ks4 number of pupil GCSE grade D passes	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
k4gcsee03	586	2003 ks4 number of pupil GCSE grade E passes	Nominal
k4gcsef03	587	2003 ks4 number of pupil GCSE grade F passes	Nominal
k4gcseg03	588	2003 ks4 number of pupil GCSE grade G passes	Nominal
k4gcsesaa03	589	2003 ks4 number of pupil short GCSE passes at A* or A	Nominal
k4gcsesac03	590	2003 ks4 number of pupil short GCSE passes at A* to C	Nominal
k4gcsesag03	591	2003 ks4 number of pupil short GCSE passes at A* to G	Nominal
k4gnvqa03	592	2003 ks4 number of pupil GNVQ or equivalent grade A* or A passes	Nominal
k4gnvqb03	593	2003 ks4 number of pupil GNVQ or equivalent grade B passes	Nominal
k4gnvqc03	594	2003 ks4 number of pupil GNVQ or equivalent grade C passes	Nominal
k4gnvqd03	595	2003 ks4 number of pupil GNVQ or equivalent grade D passes	Nominal
k4gnvqe03	596	2003 ks4 number of pupil GNVQ or equivalent grade E passes	Nominal
k4gnvqfg03	597	2003 ks4 number of pupil GNVQ or equivalent grade F or grade G passes	Nominal
k4gnvqac03	598	2003 ks4 number of pupil GNVQ or equivalent grade A* to C passes	Nominal
k4gnvqdg03	599	2003 ks4 number of pupil GNVQ or equivalent grade D to G passes	Nominal
k4higheng03	600	2003 ks4 pupil's highest English grade	Nominal
k4highmat03	601	2003 ks4 pupil's highest maths grade	Nominal
k4highsci03	602	2003 ks4 pupil's highest science grade	Nominal
k4passaa03	603	2003 ks4 pupil's total number of passes at grade A* or A	Nominal
k4passac03	604	2003 ks4 pupil's total number of passes at grades A to C	Nominal
k4passag03	605	2003 ks4 pupil's total number of passes at grades A to G	Nominal
k4passaa503	606	2003 ks4 pupil gains 5 or more passes at grade A* or grade A	Nominal
k3psag5em03	607	2003 ks4 pupil gains 5 or more passes at grades A* to G including English and maths	Nominal
k4entryg03	608	2003 ks4 pupil total GCSE/GNVQ entries	Nominal
k4psoldg03	609	2003 ks4 pupil total GCSE/GNVQ old point score	Nominal
k4pointsm03	610	2003 ks4 pupil point score in Entry Level Certificate of Education/Ceriticate of Achievement	Nominal
k4schtype03	611	2003 ks4 pupil school type	Nominal
temp	612	<none>	Scale
k45acems1	613	<none>	Scale
k45acems	614	<none>	Scale
PrimaryLast	615	Indicator of each last matching case as Primary	Ordinal
flag04	616	Flag for 2004 record	Scale
ID04	617	SPSS id 2004 (sorted in line with aphanumeric identifier)	Scale
temp2	618	<none>	Scale
pupcodeid04	619	<none>	Nominal
pupcode04	620	<none>	Nominal
pupcode05	621	<none>	Nominal
ac04	622	Census flag (January 2004)	Nominal
agegroup04	623	Pupil 2004 age group	Scale
Age04	624	Pupil age in years as at 31st August prior to start of 2003/4 school year	Scale
Agemnth04	625	Pupil age in months over and above years prior to start of 2003/4 school year	Nominal
elevenplustransfer04	626	Pupil was aged 10 in 2004 or 11 in 2005	Scale
sphase0405	627	Pupil in mainstream primary in 2004 or mainstream secondary in 2005	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
prisectrans0405	628	Pupil aged 10 2004 or 11 2004 and in mainstream pri 2004 or mainstream sec 2005	Scale
gender04	629	2004 pupil gender	Scale
sla04a	630	2004 school's maintaining LA	Nominal
sdmag1la04	631	2004 grouped (DMAG1) maintaining LA	Scale
sdmag2la04	632	2004 grouped (DMAG2) maintaining LA	Scale
sla04	633	2004 school local authority identifier	Nominal
schid04	634	2004 unique school identifier	Scale
noschool04	635	<none>	Scale
admit04	636	Admission date in 2004 record	Scale
edpcode	637	<none>	Nominal
POSTID	638	<none>	Scale
POSTCODEA	639	<none>	Nominal
ppcode04	640	Edited 2004 pupil postcode	Nominal
ppt04	641	2004 part-time pupil indicator	Scale
nursery04	642	2004 class is nursery class	Scale
boarder04	643	Pupil is boarder in 2004	Scale
ONSeth04	644	2004 pupil ethnicity, broad "ONS" type categories	Nominal
DfESgeneth04	645	2004 pupil ethnicity, general DfES categories	Nominal
DfESethdet04	646	2004 pupil ethnicity, detailed source categories	Scale
flang04	647	2004 pupil mother tongue is English or other than English	Scale
fsm04	648	2004 FSM record	Scale
sen4	649	SEN level 2004	Scale
pschlactp04	650	pupil level of SEN support 2004 at school action plus or above	Scale
SENmain04	651	Main SEN type 2004	Scale
SENsub04	652	Subsiduary SEN type 2004	Scale
pexclude04	653	Pupil permanently excluded in 12 months to Jan 2004	Scale
enrol04	654	2004 pupil enrolment status	Scale
ncyrgrp04	655	2004 national curriculum year group	Nominal
pic04	656	Pupil "looked after" on January 2004 pupil census date	Scale
piccurschl04	657	Has pupil ever been looked after while at current (2004) school?	Scale
laclea_03	658	2004 authority responsible for looked after children	Nominal
lacdmaga1	659	2004 grouped DMAG1 authority responsible for looked after children	Scale
lacdmaga2	660	2004 grouped DMAG2 authority responsible for looked after children	Scale
URN	661	2004 EDB school ID	Scale
sibob04	662	2004 school is in-borough or out-borough	Scale
sdamg104	663	2004 School dmag1 grouped LA, July 2004 EDB	Scale
sdmag204	664	2004 School dmag2 grouped LA, July 2004 EDB	Scale
sestab04	665	2004 School local DfES code. July 2004 EDB	Scale
school04	666	School name. July 2004 EDB	Nominal
spcode04	667	Edited school postcode 2004	Nominal
seasting04	668	School easting. July 2004 EDB	Scale
snorthing04	669	School northing. July 2004 EDB	Scale
sONSwardcode04	670	School ONS ward code. July 2004 EDB	Nominal
sopenclosed04	671	School open or closed. July 2004 EDB	Nominal
sreasonopen04	672	Reason for opening school. July 2004 EDB	Scale
sreasonclose04	673	Reason for closing school. July 2004 EDB	Scale
sopendate04	674	School opening date. July 2004 EDB	Scale
sclosedate04	675	School closing date. July 2004 EDB	Scale
slowage04	676	Youngest age group school caters for. July 2004 EDB	Nominal
shighage04	677	Oldest age group school caters for. July 2004 EDB	Nominal
sASClowage04	678	School's youngest ASC age group. July 2004 EDB	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
sASChighage04	679	School's oldest age ASC age group. July 2004 EDB	Scale
stotpups04	680	School total roll. July 2004 EDB	Scale
sfroll04	681	School total girls on roll. July 2004 EDB	Scale
smroll04	682	School total boys on roll. July 2004 EDB	Scale
speclass04	683	School with special classes. July 2004 EDB	Nominal
sgender04	684	School intake, boys, girls or mixed. July 2004 EDB	Scale
smaintain04	685	School maintained or independent, July 2004 EDB	Scale
sPoE04	686	School phase of education. July 2004 EDB	Nominal
simplephase04	687	Simplified school phase 2004	Scale
sphase04	688	2004 school is nursery, primary, secondary or special	Scale
sadpol04	689	School admission policy. July 2005 EDB	Scale
Adauthority	690	Is school its own admissions authority?	Scale
sToEall04	691	School ToE. July 2005 EDB	Nominal
sToEgrp04	692	School grouped ToE. July 2005. EDB	Scale
sdenom04	693	School denomination. July 2005 EDB	Scale
sgrpdenom04	694	School grouped denomination. July 2005 EDB	Scale
sdiocese04	695	VA school diocese, July 2005 EDB	Nominal
surbanrural04	696	Urban or rural school. July 2005 EDB	Nominal
sgor04	697	School GOR. July 2005 EDB	Nominal
sparlconstit04	698	School parliamentary constituency. July 2005 EDB	Nominal
sward04	699	School ward. July 2005 EDB	Nominal
sdist04	700	School district. July 2005 EDB	Nominal
slc04	701	School LSC area. July 2005 EDB	Nominal
sspecialism04	702	Specialist school status. July 2005 EDB	Nominal
scomspeclsm04	703	School (combined?) specialism. July 2005 EDB	Nominal
sspecialmes04	704	School on special measures. July 2005 EDB	Nominal
pEastingsourceg	705	pupil 2004 postcode easting	Scale
pNorthingsourceg	706	pupil 2004 postcode northing	Scale
plea04	707	pupil 2004 LEA DfES code 2004	Nominal
pLEA04b	708	pupil 2004 LEA name (SPSS autorecode)	Nominal
pdmag104	709	pupil 2004 grouped LEA codes (1)	Scale
pdmag204	710	pupil 2004 grouped LEA codes (2)	Scale
noppcodematch04	711	2004 home postcode not matched to home ward	Scale
pinoutother04	712	Pupil lives in inner London etc in 2004	Scale
pwardcode04	713	pupil 2004 ward code	Nominal
pwardspsscode04	714	pupil 2004 ward (SPSS autorecode: note wards with the the same name, but different LAs, have the same code)	Nominal
pdistrictcode04	715	pupil 2004 district/UA code	Nominal
pdistrictuaspsscode04	716	pupil 2004 district/UA (SPSS autorecode. NB: Districts with the same name, but different LAs, have the same code)	Nominal
pCountyUAcode04	717	pupil 2004 county/UA code	Nominal
pcountyUAspss04	718	pupil 2004 county/UA (SPSS autorecode)	Nominal
pCountrycode04	719	pupil 2004 country code	Nominal
pCountryname04	720	pupil 2004 country name	Nominal
pnoward04	721	<none>	Scale
pagephase04	722	pupil 2004 phase (age-based)	Scale
ibob04	723	Pupil 2004 attends in-borough or out-borough school	Scale
poaid04	724	output area spss id	Scale
pcountyco04	725	County code	Scale
pdistcode04	726	District code	Nominal
pwardcode04a	727	Ward Code	Nominal
poaseqno04	728	oaseqno	Scale
poacode104	729	Output area code	Nominal
plowersoa104	730	<none>	Nominal
LEAN04	731	<none>	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
SCHLN04	732	<none>	Scale
schoolname04	733	<none>	Nominal
stype04	734	<none>	Nominal
PTAC504	735	% 15 year olds in the school achieving 5+ A*-C grades or equiv in 2004, maintained mainstream only	Scale
PTAG504	736	% 15 year olds in the school achieving 5+ A*-G grades or equiv in 2004, maintained mainstream only	Scale
PTANYQ04	737	% of 15 year olds achieving any qualifications in 2004, maintained mainstream only	Scale
TTAPS04	738	Average GCSE and S96 point score per 15 year old pupil - adjusted, 2004, maintained mainstream only	Scale
MEAS24	739	KS2 to GCSE/GNVQ/S96 value added measure 2004, maintained mainstream only	Scale
COV24	740	Coverage indicator. % 15 year olds in KS2 to GCSE/GNVQ/S96 VA calc. 2004, maintained mainstream only	Scale
AVQUAL24	741	Av N. qualifications equiv to GCSE taken by pupils in KS2-GCSE/GNVQ/S96 VA calc, 2004, maintained mainstream only	Scale
MEAS34	742	KS3 to GCSE/GNVQ/S96 value added 2004, maintained mainstream only	Scale
COV34	743	Coverage indicator. % 15 year olds in KS3-GCSE/GNVq/S96 VA calc, 2004, maintained mainstream only	Scale
AC52001	744	% 15 year olds achieving 5+ A*-C grades or GNVQ equivalent, 2001, maintained mainstream only	Scale
AC52002	745	% 15 year olds achieving 5+ A*-C grades or GNVQ equivalent, 2002, maintained mainstream only	Scale
AC52003	746	% 15 year olds achieving 5+ A*-C grades or GNVQ equivalent, 2003, maintained mainstream only	Scale
AC52004	747	% 15 year olds achieving 5+ A*-C grades or GNVQ equivalent, 2004, maintained mainstream only	Scale
AP2001	748	% 15 year olds achieving 1+ A*-C grades or GNVQ equivalent, 2001, maintained mainstream only	Scale
AP2002	749	% 15 year olds achieving 1+ A*-C grades or GNVQ equivalent, 2002, maintained mainstream only	Scale
AP2003	750	% 15 year olds achieving 1+ A*-C grades or GNVQ equivalent, 2003, maintained mainstream only	Scale
AP2004	751	% 15 year olds achieving 1+ A*-C grades or GNVQ equivalent, 2004, maintained mainstream only	Scale
VQENT15	752	N. 15 year olds studying for relevant Vocational Qualifications or units, 2004, maintained mainstream only	Scale
PVQPA15	753	% 15 year olds achieving at least on of the qualifications/units being studied, 2004, mainstained mainstream only	Scale
k404flag	754	Flag for 2004 ks4 record	Scale
k4ac04	755	<none>	Nominal
k4age04	756	<none>	Nominal
k4mth04	757	<none>	Nominal
k4yrgrp04	758	<none>	Nominal
k4ncyg04	759	<none>	Nominal
OneAGgcsegnvq04	760	Pupil achieved 1 or more GCSE A-G grades or GNVQ equivalent, 2004	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
FiveAGgcsegnvq04	761	Pupil achieved 5 or more GCSE A-G grades or GNVQ equivalent, 2004	Scale
FiveACgcsegnvq04	762	Pupil achieved 5 or more GCSE A-C grades or GNVQ equivalent, 2004	Scale
OneAGsection9604	763	Pupil achieved 1 or more A*-G grades, section 96 qualifications, 2004	Scale
FiveACsection9604	764	Pupil achieved 5 or more A*-C grades, section 96 qualifications, 2004	Scale
FiveAGengmaths04	765	Pupil achieved 5+ A*-G grades including English and maths at GCSE/GNVQ, 2004	Scale
FiveAGengmathsci04	766	Pupil achieved 5+ A*-G grades including English, maths and science at GCSE/GNVQ, 2004	Scale
FiveACengmaths04	767	Pupil achieved 5+ A*-C grades including English and maths at GCSE/GNVQ, 2004	Scale
FiveACengmathsci04	768	Pupil achieved 5+ A*-C grades including English, maths and science at GCSE/GNVQ, 2004	Scale
FiveAGgcse04	769	Pupil achieved 5+ A-G grades - gcse only, 2004	Scale
FiveACgcse04	770	Pupil achieved 5+ A-C grades - gcse only, 2004	Scale
ptstnewe04	771	Pupil total section 96 point scores (new system), 2004	Scale
ptscnewe04	772	Pupil capped section 96 point scores (new system) 2004	Scale
ptstnewg04	773	Pupil total GCSE/GNVQ equivalised point scores (new system) 2004	Scale
ptscnewg04	774	Pupil capped GCSE/GNVQ equivalised point scores (new system) 2004	Scale
ptstoldg04	775	Pupil total GCSE/GNVQ equivalised point scores (old system) 2004	Scale
ptstoldc04	776	Pupil capped GCSE/GNVQ equivalised point scores (old system) 2004	Scale
ks2gvain04	777	Pupil average ks2 point score is input measure to ks2 to 2004 ks2 value added	Nominal
ks3gvain04	778	Pupil average ks3 point score is input measure to ks3 to 2004 ks4 value added	Scale
va2newe04	779	Pupil value added score ks2 to 2004 section 96 quals using new scoring system	Scale
va2newg04	780	Pupil value added score ks2 to 2004 GCSE/GNVQ using new scoring system	Scale
va2oldg04	781	Pupil value added score ks2 to 2004 GCSE/GNVQ using old scoring system	Scale
va3newe04	782	Pupil value added score ks3 to 2004 Section 96 quals using new scoring system	Scale
va3newg04	783	Pupil value added score ks3 to 2004 GCSE/GNVQ using new scoring system	Scale
va3oldg04	784	Pupil value added score ks2 to 2004 GCSE/GNVQ using old scoring system	Scale
k4ver04	785	<none>	Nominal
k4stype04	786	<none>	Nominal
k104flag	787	Flag for 2004 ks1 record	Nominal
k1ac104	788	<none>	Nominal
k1age104	789	<none>	Nominal
k1mth104	790	<none>	Nominal
k1lea104	791	<none>	Scale
k1estab104	792	<none>	Scale
k1gend104	793	<none>	Nominal
k1urn104	794	<none>	Scale
k1schid104	795	<none>	Scale
k1read104	796	ks1 2004 reading task, not carried out in trial schools	Nominal
k1comp104	797	ks1 2004 reading test, not carried out in trial schools	Nominal
k1writ104	798	ks1 2004 writing test, not carried out in trial schools	Nominal
k1math104	799	ks1 2004 maths task/test, not carried out in trial schools	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
k1engta104	800	ks1 2004 English overall teacher assessment, possibly not carried out in trial schools	Nominal
k1matta104	801	ks1 2004 maths overall teacher assessment, carried out in all schools	Nominal
k1scita104	802	ks1 2004 science overall teacher assessment, carried out in all schools	Nominal
k1trial104	803	Is school trialling the "no tests and fewer TAs" approach in 2004?	Nominal
k1rl2104	804	ks1 2004 achieved level 2 or above in Reading	Nominal
k1wl2104	805	ks1 2004 achieved level 2 or above in Writing	Nominal
k1ml2104	806	ks1 2004 achieved level 2 or above in mathematics	Nominal
k1sl2104	807	ks1 2004 achieved level 2 or above in science	Nominal
coded04k1read	808	Coded 2004 k1 Reading (Y/N & missing data = not at level 2+)	Scale
coded04k1write	809	Coded 2004 k1 Writing (Y/N & missing data = not at level 2+)	Scale
coded04k1maths	810	Coded 2004 k1 maths (Y/N & missing data = not at level 2+)	Scale
coded04k1sci	811	Coded 2004 k1 science (Y/N & missing data = not at level 2+)	Scale
k1rps104	812	ks1 2004 Reading total point score	Nominal
k1wps104	813	ks1 2004 Writing total point score	Nominal
k1mps104	814	ks1 2004 maths overall total point score	Nominal
k1sp104	815	ks1 2004 science overall total point score	Nominal
k1rer104	816	ks1 2004 eligible result for reading	Nominal
k1wer104	817	ks1 2004 eligible result for writing	Nominal
k1mer104	818	ks1 2004 eligible result for maths	Nominal
k1ser104	819	ks1 2004 eligible result for science	Nominal
k1stype104	820	ks1 2004 school type code	Nominal
k1sdesc104	821	ks1 2004 school type	Nominal
k1ver104	822	ks1 version	Nominal
k204flag	823	Flag for 2004 ks2 record	Scale
k2ac04	824	<none>	Nominal
k2age04	825	<none>	Nominal
k2mth04	826	<none>	Nominal
k2lea04	827	<none>	Scale
k2estab04	828	<none>	Scale
k2gend04	829	<none>	Nominal
k2urn04	830	<none>	Scale
k2schid04	831	<none>	Scale
k2eta04	832	ks2 2004 English teacher assessment	Nominal
k2mtta04	833	ks2 2004 maths teacher assessment	Nominal
k2sta04	834	ks2 2004 science teacher assessment	Nominal
k2leve04	835	k2 2004 English test level	Nominal
k2levm04	836	k2 2004 mathematics test level	Nominal
k2levs04	837	K2 2004 science test level	Nominal
k204eta4p	838	k2 2004 English TA at level 4 plus	Scale
k204ett4p	839	k2 2004 English TT at level 4 plus	Scale
k204mta4p	840	k2 2004 maths TA at level 4 plus	Scale
k204mtt4p	841	k2 2004 maths TT at level 4 plus	Scale
k204sta4p	842	k2 2004 science TA at level 4 plus	Scale
k204stt4p	843	k2 2004 science TT at level 4 plus	Scale
k2levsa04	844	k2 2004 science test level	Nominal
k2tote04	845	ks2 2004 total English test mark	Nominal
k2tiere04	846	ks2 2004 English tier - paper sat by pupil	Nominal
k2maine04	847	ks2 2004 main English paper level - same as final test level unless pupil achieved level 6 in extnsn paper (no cases of this)	Nominal
k2totm04	848	ks2 2004 total maths mark	Nominal
k2tierm04	849	ks2 2004 Maths tier - paper sat by pupil	Nominal

Appendix 1. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
k2mainm04	850	ks2 2004 maths level from main paper - same as final test level unless pupil achieved level 6 in extnsn paper (no cases of this)	Nominal
k2tots04	851	ks2 2004 science total mark (sum of marks for papers A and B)	Nominal
k2tiers04	852	ks2 2004 science paper (tier) sat by pupil (check data)	Nominal
k2mains04	853	ks2 2004 main science paper level - same as final test level unless pupil achieved level 6 in extnsn paper (no cases of this)	Nominal
k2vain04	854	ks2 2004 pupils ks1 average point score (VA input) based on final task test level in each subject	Scale
k2vaout04	855	ks2 2004 average point score (VA output) based on final test level in each subject	Nominal
k2medps04	856	ks2 2004 median point score for pupils with same or similar ks1 average point score	Nominal
k2valas04	857	ks2 2004 pupil VA score, difference, pupil's actual ks2 av point score and median ks2 score for pupils with similar ks1 points	Scale
k2schrs04	858	ks2 2004 pupil eligible for inclusion in school's performance tables (Y=yes)	Nominal
k2lears04	859	ks2 2004 pupil eligible for inclusion in LA's performance tables (Y=yes)	Nominal
k2natrs04	860	ks2 2004 pupil eligible for inclusion in national performance tables (Y=yes)	Nominal
k2lev4e04	861	ks2 2004 achieved level 4 in English	Nominal
k2lev4m04	862	ks2 2004 achieved level 4 in maths	Nominal
k2lev4s04	863	ks2 2004 achieved level 4 in science	Nominal
coded04k2Eng	864	Coded 2004 k2 English level 4+ (Y/N & missing data = not at level 4+)	Scale
coded04k2maths	865	Coded 2004 k2 maths level 4+ (Y/N & missing data = not at level 4+)	Scale
coded04k2sci	866	Coded 2004 k2 science level 4+ (Y/N & missing data = not at level 4+)	Scale
k2lev5e04	867	ks2 2004 achieved level 5 in English	Nominal
k2lev5m04	868	ks2 2004 achieved level 5 in maths	Nominal
k2lev5s04	869	ks2 2004 achieved level 5 in science	Nominal
k2totps04	870	ks2 2004 total point score	Nominal
k2slden04	871	ks2 2004 number of subjects contributing to average point score at school and LEA level	Nominal
k2nden04	872	ks2 2004 number of subjects contributing to average point score at national level	Nominal
k2stype04	873	k2 2004 school type	Nominal
k304flag	874	Flag for 2004 ks3 record	Scale
k3ac04	875	<none>	Nominal
k3_age04	876	<none>	Nominal
k3_mth04	877	<none>	Nominal
k3_lea04	878	<none>	Scale
k3_estab04	879	<none>	Scale
k3_gend04	880	<none>	Nominal
k3_urn04	881	<none>	Scale
k3_schid04	882	<none>	Scale
k3_engta04	883	ks3 2004 English TA level	Nominal
k3_matta04	884	ks3 2004 maths TA level	Nominal
k3_scita04	885	ks3 2004 science TA level	Nominal
k3_leve04	886	ks3 2004 English test level	Nominal
k3_erftl04	887	ks3 2004 English reading test level	Nominal
k3_ewftl04	888	ks3 2004 English writing test level	Nominal
k3_levm04	889	ks3 2004 maths test level	Nominal
k3_levs04	890	ks3 2004 science test level	Nominal
k304etm	891	ks3 2004 total English test mark	Nominal
k304mtm	892	ks3 2004 total maths test mark	Nominal
k304stm	893	ks3 2004 total science test mark	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
k304erm	894	ks3 2004 English reading test mark (pupils at levels 4 to 7)	Nominal
k304ewm	895	ks3 2004 English writing test mark (pupils at levels 4 to 7)	Nominal
k304ewsm	896	ks3 2004 English Shakespeare writing test mark (pupils at levels 4 to 7)	Nominal
k304ertm	897	ks3 2004 English total reading test mark	Nominal
k304ewtm	898	ks3 2004 English total writing test mark	Nominal
k304tote	899	ks3 2004 total English test marks (reading plus writing)	Nominal
k304tiere	900	ks3 2004 English paper tier	Nominal
k304maine	901	ks3 2004 English main paper level (same as test unless pupil reached 8 in extension test)	Nominal
k304marit	902	ks3 2004 mark given in mental arithmetic paper of maths main test	Nominal
k304totm	903	ks3 2004 total maths test mark	Nominal
k304tierm	904	ks3 2004 maths paper (tier) taken	Nominal
k304mainm	905	ks3 2004 maths main paper level (same as test unless pupil reached 8 in extensionn test)	Nominal
k304tots	906	ks3 2004 total science test marks	Nominal
k304tiers	907	ks3 2004 science paper (tier) taken	Nominal
k304mains	908	ks3 2004 science main paper level (same as test unpless pupil reached 8 in extension test)	Nominal
k304vain	909	ks3 2004 ks2 average point score from final test levels (VA input)	Nominal
k304vaout	910	ks3 2004 average point score from final test levels (VA output)	Nominal
k304medps	911	ks3 2004 median ks3 average point score for pupils with same/similar ks2 average point score	Nominal
k304valas	912	ks3 2004 value added score - difference pupil's actual ks3 av point score and median for pupils with similar ks2 points	Nominal
k304schrs	913	ks3 2004 Y=pupil enligible for inclusion in school's performance tables	Nominal
k304lears	914	ks3 2004 Y=pupil enligible for inclusion in LA's performance tables	Nominal
k304natrs	915	ks3 2004 Y=pupil enligible for inclusion in national performance tables	Nominal
k304lev5e	916	ks3 2004 pupil achieved level 5 or above in ks3 English	Nominal
k304lev5m	917	ks3 2004 pupil achieved level 5 or above in ks3 maths	Nominal
k304lev5s	918	ks3 2004 pupil achieved level 5 or above in ks3 science	Nominal
coded04k3eng	919	Coded ks3 English 2004 level 5+ (missing = not at level 5+)	Nominal
coded04k3maths	920	Coded ks3 mathematics level 5+ (missing = not at level 5+)	Nominal
coded04k3sci	921	Coded ks3 science level 5+ (missing = not at level 5+)	Nominal
k304totps	922	ks3 2004 total point score	Nominal
k304slden	923	ks3 2004 number of subjects contributing to average point score at school and LA level	Nominal
k304nden	924	ks3 2004 number of subjects contributing to average point score at national level	Nominal
k304stype	925	ks3 2004 school type	Nominal
k304ver	926	ks3 2004 file version	Nominal
Postcode	927	<none>	Nominal
plowersoa	928	<none>	Nominal
LA_CODE	929	<none>	Nominal
LA_	930	<none>	Nominal
GOR_CODE	931	<none>	Nominal
IMDscore	932	Index of multiple deprivation score	Scale
IMDrank	933	<none>	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
IMDIncomescore	934	<none>	Scale
IMDincomerank	935	<none>	Scale
IMDemploymentscore	936	<none>	Scale
IMDemploymentrank	937	<none>	Scale
IMDhealthscore	938	<none>	Scale
IMDhealthrank	939	<none>	Scale
IMDedscore	940	<none>	Scale
IMDedrank	941	<none>	Scale
IMDbarrerscore	942	<none>	Scale
IMDbarrierrank	943	<none>	Scale
IMDcrimescore	944	<none>	Scale
IMDcrimerank	945	<none>	Scale
IMDenvirscore	946	<none>	Scale
IMDenvirrank	947	<none>	Scale
IMDGOR	948	<none>	Nominal
IMDdistricts	949	<none>	Nominal
paycheckflag	950	January 2006 Paycheck file flag (not April 2006)	Scale
Paycheckpcode	951	<none>	Nominal
DELETEDFLAG	952	<none>	Nominal
LARGEUSERFLAG	953	<none>	Nominal
meantext1	954	<none>	Nominal
meantext	955	<none>	Nominal
k4oldpointsgrp04	956	2004 GCSE old point score group	Scale
paycheckmeangroup	957	2005 Paycheck data at postcode level. Income ranges.	Scale
paycheckmeangroupA	958	2005 Paycheck 6 income groups	Scale
paycheckMEAN	959	<none>	Scale
paycheckMEDIAN	960	<none>	Scale
paycheckMODE	961	<none>	Scale
paycheckTOTALHOUSEHOLDS	962	<none>	Scale
Continuity	963	Years for which pupil has a record	Scale
flag05	964	<none>	Scale
continuity0405	965	<none>	Scale
ppcode0405	966	2004 2005 home postcode continuity and discontinuity	Scale
pscode0405	967	School continuity 2004 2005	Scale
lacontinuity0405	968	Home and school LA continuity 2004 to 2005	Scale
homeschool0405	969	Home school stability mobility 2004 2005	Scale
notendofphase0405	970	Pupil not end of phase in 2004 or start of phase in 2005. No Harrow	Scale
ID05	971	2005 SPSS ID, in same sequence as pupil code	Nominal
ac_05	972	<none>	Nominal
temp1	973	<none>	Scale
pupcode05a	974	<none>	Nominal
Londonpupil05	975	Pupil lives in or attends school in London	Scale
age05b	976	Pupil age at 31st August 2004	Scale
DOB05	977	<none>	Scale
borndd05	978	2005 roll, day born	Nominal
bornmm05	979	2005 roll, month of birth	Nominal
bornyyyy05	980	2005 roll, year born	Nominal
age05	981	2005 roll, age in whole years	Scale
month_05	982	<none>	Nominal
gen05	983	<none>	Scale
gend_05	984	<none>	Nominal
schlid05	985	<none>	Scale
noschool05	986	No record of 2005 school	Scale
tempschlid05	987	<none>	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
sla05	988	<none>	Scale
estab_05	989	<none>	Scale
laest_05	990	<none>	Scale
urn_05	991	<none>	Scale
new05	992	Pupil admitted after 2004	Scale
loftime05	993	2005 pupil lrength of time on roll in years	Scale
aug05	994	31st August 2005	Scale
census05	995	2005 date of pupil census	Scale
admit05	996	2005 roll, date pupil admitted	Scale
ddadmit05	997	2005 pupil day admitted	Nominal
mmadmit05	998	2005 pupil month admitted	Nominal
yyadmit05	999	2005 pupil year admitted	Nominal
post_05	1000	<none>	Nominal
ppcode05	1001	Edited 2005 pupil home postcode	Nominal
ppcode05a	1002	<none>	Nominal
pti_05	1003	<none>	Nominal
cti_05	1004	<none>	Nominal
pboard05	1005	<none>	Nominal
pethcode05	1006	<none>	Scale
pethsc05	1007	<none>	Nominal
pflang05	1008	<none>	Nominal
fsm05	1009	2005 FSM record	Nominal
psen05	1010	<none>	Nominal
penrol_05	1011	<none>	Nominal
pncyr05	1012	<none>	Nominal
pcare05	1013	<none>	Nominal
pcsch_05	1014	<none>	Nominal
pcauth_05	1015	<none>	Scale
pconn05	1016	<none>	Nominal
psen105	1017	2005 pupil main SEN type	Nominal
psen205	1018	2005 pupil subsidiary SEN type	Nominal
ppa05	1019	<none>	Nominal
pAlev_05	1020	<none>	Nominal
pGCSE_05	1021	<none>	Nominal
pGNVQ_05a	1022	<none>	Nominal
PGNVQ_05	1023	<none>	Nominal
pNVQ_05	1024	<none>	Nominal
pOther_05	1025	<none>	Nominal
ppei_05	1026	<none>	Nominal
poa_05	1027	<none>	Nominal
psoa_05	1028	<none>	Nominal
pidaci_05	1029	<none>	Scale
prank_05	1030	<none>	Scale
ptemp05	1031	<none>	Scale
peasting05	1032	pupil 2005 postcode easting	Scale
pnorthing05	1033	pupil 2005 postcode northing	Scale
plea05	1034	pupil LEA DfES code 2005	Nominal
pleaname05	1035	pupil LEA name 2005	Nominal
pdmag105	1036	pupil 2005 Grouped LEA codes (1)	Scale
pdmag205	1037	pupil 2005 Grouped LEA codes (2)	Scale
pnewLAcode05	1038	pupil 2005 'new' LA codes	Scale
dummy	1039	<none>	Nominal
pinoutother05	1040	<none>	Scale
noppcodematch05	1041	<none>	Scale
pwardcode05	1042	Ward code 2005	Nominal
pWardname05	1043	Ward name 2005	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
pwardspsscode05	1044	2005 Ward (SPSS autorecode: note wards with the same name, but different LAs, have the same code)	Nominal
pdistrictcode05	1045	District/UA code 2005	Nominal
pdistrictname05	1046	District/UA name 2005	Nominal
pdistrictuaspsscode05	1047	2005 district/UA (SPSS autorecode. NB: Districts with the same name, but different LAs, have the same code)	Nominal
pCountyUAcode05	1048	County/UA code 2005	Nominal
pCountyUAname05	1049	County/UA name 2005	Nominal
pcountyUAspss05	1050	2005 county/UA (SPSS autorecode)	Nominal
pCountrycode05	1051	Country code 2005	Nominal
pCountryname05	1052	Country name 2005	Nominal
sURNA05	1053	2005 EDB school ID	Scale
sCOPYLEAB05	1054	<none>	Scale
sLEAB05	1055	<none>	Nominal
snewlacode05a	1056	<none>	Scale
sdmag105	1057	<none>	Scale
sdmag205	1058	<none>	Scale
sunique05a	1059	School unique id. July 2005 EDB	Scale
sestab05a	1060	School local DfES code. July 2005 EDB	Scale
school05a	1061	School name. July 2005 EDB	Nominal
spcode05a	1062	Edited school postcode 2005	Nominal
seasting05a	1063	School easting. July 2005 EDB	Scale
snorthing05a	1064	School northing. July 2005 EDB	Scale
sONSwardcode05a	1065	School ONS ward code. July 2005 EDB	Nominal
sopenclosed05a	1066	School open or closed. July 2005 EDB	Nominal
sreasonopen05a	1067	Reason for opening school. July 2005 EDB	Scale
sreasonclose05a	1068	Reason for closing school. July 2005 EDB	Scale
sopendate05a	1069	School opening date. July 2005 EDB	Scale
sclosedate05a	1070	School closing date. July 2005 EDB	Scale
slowage05a	1071	Youngest age group school caters for. July 2005 EDB	Nominal
shighage05a	1072	Oldest age group school caters for. July 2005 EDB	Nominal
sASClowage05a	1073	School's youngest ASC age group. July 2005 EDB	Scale
sASChighage05a	1074	School's oldest age ASC age group. July 2005 EDB	Scale
stotpups05a	1075	School total roll. July 2005 EDB	Scale
sfroll05a	1076	School total girls on roll. July 2005 EDB	Scale
smroll05a	1077	School total boys on roll. July 2005 EDB	Scale
sAPP_SPEC_PUPILSa	1078	<none>	Scale
speclass05a	1079	School with special classes. July 2005 EDB	Nominal
sboarders05a	1080	Boarders. July 2005 EDB	Nominal
snursery05a	1081	Nursery classes. July 2005 EDB	Nominal
scapacitya	1082	School capacity. July 2005 EDB	Scale
sgender05a	1083	School intake, boys, girls or mixed. July 2005 EDB	Scale
smaintain05a	1084	School maintained or independent, July 2005 EDB	Scale
sPoE05a	1085	DfES School phase of education. July 2005 EDB	Nominal
sphase05	1086	Simplified school phase 2005	Scale
Mainstream05	1087	2005 school is a mainstream or a special school	Scale
sadpol05a	1088	School admission policy. July 2005 EDB	Scale
sToEall05a	1089	School ToE. July 2005 EDB	Nominal
sToEgrp05a	1090	School grouped ToE. July 2005. EDB	Scale
simpletoe05	1091	Simplified ToE School Community/VC or VA/foundation/CTC or Academy	Scale
sdenom05a	1092	School denomination. July 2005 EDB	Scale
sgrpdenom05a	1093	School grouped denomination. July 2005 EDB	Scale
sdiocese05a	1094	VA school diocese, July 2005 EDB	Nominal
surbanrural05a	1095	Urban or rural school. July 2005 EDB	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
sgor05a	1096	School GOR. July 2008 EDB	Nominal
sparlconstit05a	1097	School parliamentary constituency. July 2005 EDB	Nominal
sward05a	1098	School ward. July 2005 EDB	Nominal
sdist05a	1099	School district. July 2005 EDB	Nominal
sttwa05a	1100	School travel to work area. July 2005 EDB	Nominal
slc05a	1101	School LSC area. July 2005 EDB	Nominal
sspecialism05a	1102	Specialist school status. July 2005 EDB	Nominal
scomspeclsm05a	1103	School (combined?) specialism. July 2005 EDB	Nominal
sspecialmes05a	1104	School on special measures. July 2005 EDB	Nominal
seaz05a	1105	School EAZ. July 2005 EDB	Nominal
sbeacon05a	1106	Beacon school. July 2005 EDB	Nominal
seic05a	1107	School EiC. July 2005 EDB	Nominal
eicgrp05a	1108	School EiC group. July 2005 EDB	Nominal
sEiCLS05a	1109	School is EiC City Learning Centre. July 2005 EDB	Nominal
sEiCAZ05a	1110	School EiC Action Zone. July 2005 EDB	Nominal
sfreshstart05a	1111	Fresh Start school. July 2005 EDB	Nominal
straining05a	1112	Training school. July 2005 EDB	Nominal
searlyex05a	1113	Early (Years) Excellence Centre. July 2005 EDB	Nominal
spfi05a	1114	School part of PFI. July 2005 EDB	Nominal
s6thform05a	1115	School has 6th form. July 2005 EDB	Nominal
searlytype05a	1116	Type of early years provision. July 2005 EDB	Nominal
sofsteinspec05a	1117	Last OfSTED inspecation date. July 2005 EDB	Scale
filter_$	1118	Londonpupil05=1 (FILTER)	Scale
pdamg105	1119	<none>	Scale
pdamg205	1120	<none>	Scale
pnewLAcodes05	1121	<none>	Scale
pneLAcode05	1122	<none>	Scale
pinoutother	1123	<none>	Scale
noschoolid04	1124	<none>	Scale
lacontinuity	1125	<none>	Scale
housingflag	1126	<none>	Scale
POSTCODEB	1127	<none>	Nominal
PC_AREA	1128	<none>	Nominal
POSTSECT	1129	<none>	Nominal
AVG_DET	1130	Detached housing average price 1st quarter 2002 to end of 2nd quarter 2003	Scale
AVG_S_DET	1131	Semi-detached housing average price 1st quarter 2002 to end of 2nd quarter 2003	Scale
AVG_TER	1132	Terraced housing average price 1st quarter 2002 to end of 2nd quarter 2003	Scale
AVG_FLAT_M	1133	Flat or maisonette average price 1st quarter 2002 to end of 2nd quarter 2003	Scale
AVG_OVERAL	1134	Housing average overall price 1st quarter 2002 to end of 2nd quarter 2003	Scale
DENTISTS	1135	Dentists - number in postcode sector 2001	Nominal
GPS	1136	GPs - number in postcode sector 2001	Nominal
DIST_NHS	1137	Hospitals - nearest in kilometers	Scale
DIST_GREEN	1138	Open green space - nearest in kilometers	Scale
GREEN_LESS	1139	Open green sites - number within 1 kilometer radius	Scale
DIST_BR_LU	1140	Station - nearest BR or LU station in kilometers	Scale
NO_BRLU_LE	1141	Station - number of BR/LU stations within 1 kilometer radius	Scale
STATIONS	1142	Station - number BL/LU in postcode sector	Nominal
TT_PT2001	1143	2001 public transport travel time to central London	Scale
TT_HW2001	1144	2001 road travel time to central London	Scale
DIST_SCHOO	1145	Secondary school - nearest in kilometers	Scale
NO_SCHOOLS	1146	Secondary (?) schools within 2 kilometer radius (N)	Nominal
SCHOOLS	1147	Secondary (?) schools in the postcoder sector (N)	Nominal

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
KS3_APS	1148	KS3 average point score of schools within n2 km radius	Scale
PCNTY115AC	1149	K4 % 5A*-C withn 2 km radius	Scale
ALL_PEOPLE	1150	Total resident population in postcode secot in 2001	Scale
WHITE	1151	Percentage White in 2001	Scale
ECOLY_ACTI	1152	Age 16-74 percentage economically active 2001	Scale
DETACHED	1153	Household spaces detached 2001 - percentage	Nominal
SEMI_DET	1154	Household spaces semi-detached 2001 - percentage	Scale
TERRACED	1155	Household spaces terraced 2001 - percentage	Scale
FLAT_MAI_A	1156	Household spaces flats or maisonettes 2001 - percentage	Scale
OWNER_OCC	1157	Percentage of households owner occupiers 2001	Scale
SOCIAL_REN	1158	Percentage of households socially renting 2001	Scale
PVT_RENTED	1159	Percentage of households privately renting	Scale
AVG_HLD_SI	1160	Household size (average) 2001	Scale
AVG_RMS_PE	1161	N. Room per household (average) 2001	Scale
OVERCROWDI	1162	% households with occupancy rate of minus 1 or less 2001	Scale
ONE_PER_HL	1163	% households 1 person 2001	Scale
COUPLE_DEP	1164	% households with dependent children 2001	Scale
IS_1998	1165	N. Income Support claimants 1998	Scale
IS_1999	1166	N. Income Support claimants 1999	Scale
IS_2000	1167	N. Income Support claimants 2000	Scale
SO2_T_A	1168	SO2 emisions in tons (1999?)	Scale
NOX_T_A	1169	Nitrus Oxide emissions in tonnes (1999?)	Scale
CO2_T_A	1170	CO2 emissions in tonnes (1999?)	Scale
PM10_T_A	1171	Particulate matters emissions (10 millionth of 1 mm) in tommes (1999?)	Scale
flag02030405a	1172	LPD record 2002 to 2005	Scale
flag030405	1173	Pupils roll record in 2003 2003 and 2005	Scale
age904	1174	Pupils aged 9 in 2004	Scale
age1004	1175	Pupils aged 10 in 2004	Scale
age1404	1176	Pupils aged 14 in 2004	Scale
age5to904	1177	Pupils aged 5 to 9 in 2004	Scale
age11to1404	1178	Pupils aged 11 to 14 in 2004	Scale
notendofphase0304	1179	Pupils not end of phase in 2003, not attending a middle school in either 2003 or 2004 and with matchable pcode both years	Scale
childmobility0304	1180	Child mobility between 2003 and 2004	Scale
cmob0304onroll05	1181	Child mobility 2003 to 2004 and roll status in 2005	Scale
countemp	1182	<none>	Scale
rollstatus030405	1183	On roll 20003 2004 2005	Scale
returners	1184	Pupils with intermitent roll records 2002 to 2005	Scale
oneyearonly0205	1185	Pupils with and LPD record for one year only 2002 to 2005	Scale
LPDcontinuity02030405	1186	Continuity in the LPD record 2002 to 2005	Scale
odd	1187	<none>	Scale
flag0405	1188	Pupil on roll in 2004 and 2005	Scale
scountsum	1189	Number aged 10, each school in 2004	Scale
sk204ett4psum	1190	Number at 4+ k2 English TT 2004 in each school	Scale
sk204mtt4psum	1191	Number at 4+ k2 maths TT 2004 in each school	Scale
sk204stt4psum	1192	Number at 4+ k2 science TT 2004 in each school	Scale
sk2totps04_mean	1193	Average 2004 total pupil k2 point score in each school	Scale
squartk2points04	1194	School 2004 k2 quartile, pupil average point score	Scale
spcettlev4p	1195	% reaching level 4+ in English TT, 2004 in each school	Scale
spcmttlev4p	1196	% at level 4+ maths TT, 2004 in each school	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
spcsttlev4p	1197	% at level 4p science TT, 2004 in each school	Scale
squartk2ett04	1198	School 2004 quartile k2 English TT % level 4+ (pupils aged 10)	Scale
squartk2mtt04	1199	School 2004 quartile k2 maths TT % level 4+ (pupils aged 10)	Scale
squartk2stt04	1200	School 2004 quartile k2 science TT % level 4+ (pupils aged 10)	Scale
sptsnewe04mean	1201	2004 average total S96 points in the school (15 year olds)	Scale
squartsec96points	1202	School quartiles 2004 Sec 96 total points (pupils aged 15)	Scale
k4flag05	1203	<none>	Scale
k4pmr05	1204	<none>	Nominal
k4ac05	1205	<none>	Nominal
k4age05	1206	<none>	Nominal
k4mth05	1207	<none>	Nominal
k4dob05	1208	<none>	Nominal
k4yrgrp05	1209	<none>	Nominal
k4ncyg05	1210	<none>	Nominal
k4pem05	1211	<none>	Nominal
k4latc05	1212	<none>	Scale
k4laest05	1213	<none>	Scale
k4nume05	1214	<none>	Scale
k4amend05	1215	<none>	Nominal
k4la05	1216	<none>	Scale
k4estab05	1217	<none>	Scale
k4gend05	1218	<none>	Nominal
k4land05	1219	<none>	Nominal
hospind05	1220	<none>	Nominal
earlyte05	1221	<none>	Nominal
norflage05	1222	<none>	Nominal
schres05	1223	<none>	Nominal
lares05	1224	<none>	Nominal
natres05	1225	<none>	Nominal
schnor05	1226	<none>	Nominal
lanor05	1227	<none>	Nominal
natnor05	1228	<none>	Nominal
fiveac05	1229	<none>	Nominal
level205	1230	<none>	Nominal
fiveag05	1231	<none>	Nominal
level105	1232	<none>	Nominal
oneag05	1233	<none>	Nominal
anylev105	1234	<none>	Nominal
ANYPASS05	1235	<none>	Nominal
LEV2EM05	1236	<none>	Nominal
LEV2FEM05	1237	<none>	Nominal
LEV1FEM05	1238	<none>	Nominal
ptstnewe05	1239	2005 GCSE and equivalent point scores	Scale
pointquart05	1240	2005 uncapped GCSE and equivalent point score quartiles	Scale
ptscnewe05	1241	2005 capped GCSE and equivalent point scores	Scale
ptstnewg05	1242		Scale
ptscnewg05	1243	<none>	Scale
gcsesac05	1244	<none>	Nominal
gcsesag05	1245	<none>	Nominal
higheng05	1246	<none>	Nominal
highmat05	1247	<none>	Nominal
highsci05	1248	<none>	Nominal
passaaa05	1249	<none>	Scale
passac05	1250	<none>	Scale

Appendix. Variable list merged 2002 to 2005 LPD - continued

Variable	Position	Label	Measurement Level
passag05	1251	<none>	Scale
passelq05	1252	<none>	Nominal
passksl105	1253	<none>	Nominal
passksl205	1254	<none>	Nominal
aoraa505	1255	<none>	Nominal
ag5em05	1256	<none>	Nominal
ag5ems05	1257	<none>	Nominal
ac5em05	1258	<none>	Nominal
ac5ems05	1259	<none>	Nominal
levl1em05	1260	<none>	Nominal
lev1ems05	1261	<none>	Nominal
levl2em05	1262	<none>	Nominal
lev2ems05	1263	<none>	Nominal
acems05	1264	<none>	Nominal
psnewg05	1265	<none>	Nominal
psnewe05	1266	<none>	Nominal
psoldg05	1267	<none>	Scale
pschlactp	1268	<none>	Scale
fsmsum	1269	<none>	Scale
tempschool04	1270	<none>	Scale
fsm0204	1271	<none>	Scale
fsm0205	1272	FSM record 2002 to 2005	Scale
rollstatus0205	1273	Roll status 2002 2005	Scale
LPDpupil05	1274	Did pupil live in London or attend a London school in 2005	Scale
fsmflag02	1275	FSM flag 2002 (0=not entitled or not on roll in 2002)	Scale
fsmflag04	1276	FSM flag 2004 (0=not entitled or not on roll in 2004)	Scale
fsmflag05	1277	FSM flag 2005 (0=not entitled or not on roll in 2005)	Scale
fsmflag03	1278	FSM flag 2003 (0=not entitled or not on roll in 2003)	Scale
fsmflagsum0205	1279	FSM record 02 05 (0=not entitled or not on roll)	Scale

Source: 2002 to 2005 LPD

Regular Briefings from the GLA Data Management and Analysis Group - DMAG

Some recent DMAG Briefings:

2008-04	Council Tax Analysis	Elizabeth Williams
2008-05	A Profile of Londoners by Country of Birth	Lorna Spence
2008-08	Greater London Authority Constituency Profiles	Elizabeth Williams & Caroline Hall
2008-09	Family Resources Survey 2005/06: Results for London	Lovedeep Vaid
2008-10	London Borough Migration 2001-06	John Hollis
2008-15	2001 Census Profiles: Black Caribbeans in London	Richard Cameron
2008-17	Lone Parents on Income Support by Ethnic Group	Lovedeep Vaid
2008-18	Schools Key Facts and Trends 2003-07	Shen Cheng
2008-19	2008 Elections results summary	Gareth Piggott
2008-21	Indices of Deprivation 2007: A London perspective	Rachel Leeser
2008-22	London Ward Level Summary Measures for the Indices of Deprivation 2007	Rachel Leeser
2008-24	2001 Census: Ethnic Group Migration Structures (as used in Model)	Baljit Bains/Ed Klodawski
2008-26	London Council By-Election Results, May 2006 to July 2008	Gareth Piggott
2008-27	Social Selection, Social Sorting and Education; "Missing" Children	David Ewens
2008-28	Summary of Social Trends 2008	Elizabeth Williams
2008-29	Children in Benefit Families 2007	Lovedeep Vaid
2008-30	Londoners and the Labour Market: Key Facts	Lorna Spence
2008-31	Child Poverty In London: 2008 Update	Social Exclusion Data Team
2008-33	Paycheck 2008	Lovedeep Vaid
2008-34	Background Poverty Profiles	Lovedeep Vaid
2009-01	Claimant Count Model 2009: Technical Note	Social Exclusion Data Team
2009-02	GLA 2008 Round Demographic Projections	John Hollis/Jessica Chamberlain
2009-03	Greater London Demographic Review 2007	John Hollis
2009-04	Census Information Note 2009-1	Eileen Howes
2009-05	Census Information Note 2009-2	Eileen Howes
2009-06	2001 Census Consortium and Information Scheme	Eileen Howes
2009-07	2009 European Election Results for London	Gareth Piggott
2009-08	GLA 2008 Round Ethnic Group Population Projections	Ed Klodawski

A full list of DMAG Briefings is available to internal customers through the GLA Intranet; otherwise please contact dmag.info@london.gov.uk A CD containing PDF versions of the Briefings, or hard copies, can be provided.