



# CLS Cohort Studies

## Data Note 2011/1

Centre for  
Longitudinal  
Studies

Deriving Highest Qualification  
in NCDS and BCS70

Brian Dodgeon and Samantha Parsons

**Centre for Longitudinal Studies**

Institute of Education

20 Bedford Way

London WC1H 0AL

Tel: 020 7612 6860

Fax: 020 7612 6880

Email [cls@ioe.ac.uk](mailto:cls@ioe.ac.uk)

Web <http://www.cls.ioe.ac.uk>

# Highest Educational Qualification in NCDS and BCS70

This Data Note explains the derivation of variables on Highest Educational Qualification obtained by cohort members in the National Child Development Study (NCDS) and the 1970 British Cohort Study (BCS70).

## Contents

	<b>Page</b>
<b>1. Introduction .....</b>	<b>2</b>
<b>2. Reconciling academic and vocational educational levels .....</b>	<b>3</b>
<b>3. Longitudinal tracking of highest qualification level over time – NCDS .....</b>	<b>4</b>
<b>4. Longitudinal tracking of highest qualification level over time – BCS70 .....</b>	<b>5</b>
<b>Appendices .....</b>	<b>6</b>
<b>Appendix 1 Derived variable names and descriptions .....</b>	<b>6</b>
<b>Appendix 2 Syntax to derive to derive Highest Educational Qualification in NCDS4 (age 23 survey 1981 – variables already deposited with NCDS4 dataset) .....</b>	<b>8</b>
<b>Appendix 3 Syntax to derive Highest Educational Qualification in NCDS6 (age 42 survey 1999/2000), and combine with information from NCDS5 (age 33 survey 1991) .....</b>	<b>13</b>

## 1. Introduction

The full NCDS cohort were surveyed at ages 0, 7, 11, 16, 23, 33, 42, 44 (Biomedical Survey), 46 and 50.

Information on educational qualifications obtained in NCDS was gathered via:

- (a) **The NCDS Exams exercise (1978):** Details of examination performance by members of the NCDS cohort were obtained by writing to schools which study members were known to attend at the time of the 1974 follow-up (NCDS3). Schools were asked to provide details of all CSE, GCE and (in Scotland) SCE examinations entered up to 1978. They were also asked to indicate when an individual was known to have gone to study elsewhere (eg. another school, sixth-form college, FE college, etc) and might have taken more such examinations. In such cases the other institutions were also contacted and the relevant results obtained. See CLS website (<http://www.cls.ioe.ac.uk/ncds>) for more information.
- (b) **The NCDS4 Sweep (1981, age 23):** Respondents were asked to list all academic and vocational qualifications obtained since school, and apprenticeships completed. 'Highest qualification' variables were derived and deposited with the main NCDS4 dataset (see SPSS syntax in Appendix 2), but since the summary categories derived differ from those used in later sweeps consolidating all qualifications up to age 42 and beyond, these NCDS4 variables can be considered to have been superseded (except perhaps in the case of research which seeks to distinguish qualifications gained by age 23 from those gained later in life).
- (c) **The NCDS5 Sweep (1991, age 33):** Respondents were asked to list all academic and vocational qualifications obtained since school.
- (d) **The NCDS6 Sweep (1999/2000, age 42):** Respondents were asked to list all academic and vocational qualifications obtained since March 1991 (*if present at NCDS5 - otherwise since school*), and apprenticeships completed.
- (e) **The NCDS7 Sweep (2004, age 46):** Respondents were asked to list all academic and vocational qualifications obtained since last interview (*if present at NCDS6 - otherwise since March 1991*), and apprenticeships completed.
- (f) **The NCDS8 Sweep (2008, age 50):** Respondents were asked to list all academic and vocational qualifications obtained since last interview (*if present at NCDS7 - otherwise since 1<sup>st</sup> January 2000*), and apprenticeships completed.

The full BCS70 cohort were surveyed at ages 0, 5, 10, 16, 26, 30, 34 and 38.

Information on educational qualifications obtained in BCS70 was gathered via:

- (a) **The age 26 BCS70 Sweep (1996):** This was a postal survey in which respondents were asked to list all academic and vocational qualifications obtained since school.
- (b) **The age 30 BCS70 Sweep (1999/2000):** Respondents were again asked to list all academic and vocational qualifications obtained since school, but in much more detail than in the condensed categories of the 1996 survey.
- (c) **The age 34 BCS70 Sweep (2004):** Respondents were asked to list all academic and vocational qualifications obtained since they were last interviewed (i.e. not counting 1996 survey, which was postal).

**(d) The age 38 BCS70 Sweep (2008):** Respondents were asked to list all academic and vocational qualifications obtained since last interview (*if present at age 34 Sweep - otherwise since 1<sup>st</sup> January 2000*), and apprenticeships completed.

## 2. Reconciling academic and vocational educational levels

In both NCDS and BCS70 we have, for each sweep from 1999/2000 onwards, performed four derivations:

- Highest Academic level
- Highest NVQ level (academic)
- Highest NVQ level (vocational)
- Highest NVQ level (academic or vocational)

Looking first at academic qualifications, we group them into 9 categories:

- 0 No qualifications
- 1 Bad O-levels
- 2 CSE grades 2-5
- 3 Good O-levels
- 4 2+ AS levels or 1 A level
- 5 More than one A level
- 6 Diploma
- 7 Degree or PGCE
- 8 Higher degree

These value labels are very brief and inadequate descriptors of a detailed system of aggregation of all possible academic qualifications: For example, GCSEs are not mentioned here explicitly, since they replaced O levels in the 1986/87 academic year, which meant both NCDS and BCS70 members were born too early to do GCSEs at school. But they may have acquired them later, in which case they go in category 1 or 3 depending on whether grade A-C or lower). The various levels of Scottish Qualifications aren't explicitly referenced. The way in which every possible academic qualification is mapped into the categories above, is detailed in the SPSS syntax provided in Appendix 3.

Then, to facilitate comparisons between academic and non-academic qualifications, we collapse these academic categories into six levels deemed to be equivalent to the various levels of [National Vocational Qualifications](#) (NVQs):

- Academic level 1 or 2 = NVQ level 1 equivalent
- Academic level 3 or 4 = NVQ level 2 equivalent
- Academic level 5 = NVQ level 3 equivalent
- Academic level 6 or 7 = NVQ level 4 equivalent
- Academic level 8 = NVQ level 5 equivalent

This mapping was provided by the [Qualifications and Curriculum Authority](#) (QCA), which existed between October 1997 and April 2010, being superseded by Ofqual and the Qualifications and Curriculum Development Agency.

Note that the equivalence of certain academic levels with certain vocational levels is a matter of ongoing debate: at one time a (first) degree was deemed equivalent to NVQ level 5, but this was revised to NVQ level 4 by the QCA in the late 1990s.

The third of the four derived variables aggregates all the possible vocational qualifications into these same five NVQ levels, according to the [National Qualifications Framework](#) (NQF). The NQF has eight levels, of which the top

three are collapsed into NVQ level 5, the next two into NVQ level 4, and the lowest three map into the lowest three NVQ levels.

Finally, a fourth variable is produced which compares the highest NVQ level acquired with the 'NVQ-equivalent' level of qualification attained in the academic sphere, to produce a 'Highest NVQ level (academic or vocational)'.

### **3. Longitudinal tracking of highest qualification level over time - NCDS**

#### **(a) NCDS6 derived variables**

The age 33 sweep is taken as the pivotal starting point, since that is the latest sweep at which the whole cohort were asked to recap all academic and vocational courses they had been on, and all qualifications obtained, since school.

At all subsequent sweeps, respondents were only asked about qualifications obtained since the last interview, so in order to keep track of which was their highest qualification ever, an element of longitudinal data linkage is always necessary.

The first step was to derive a set of four highest educational qualification variables from the NCDS5 (1991) dataset.

Then a similar set of four derived variables was produced from the NCDS6 (1999/2000) dataset.

Finally, the NCDS5 and NCDS6 information was then merged and compared, and a further set of four variables was produced, giving the highest level from both NCDS5 and NCDS6 (see SPSS syntax in Appendix 3).

Of course, as in any longitudinal merge, there were some cohort members present in NCDS6 who were not present in NCDS5 (and vice versa). These can be identified by the NCDS6 variable DMPART. These people were asked at NCDS6 about all their qualifications *since school*, whereas those present at NCDS5 were only asked about qualifications since that last interview.

Appendix 1 shows the two sets of four variables derived for those present at NCDS6:

- Highest qualification (four types of categorisation) up to and including NCDS6
- Highest qualification (four types of categorisation) stated at the NCDS6 survey

The first four (hqaca00, hqanvq00, hqvoc00, hqnvq00) are the most useful, as they longitudinally combine all information known up to that date. The other four (hiaca00, nvqaca00, hivoc00, hinvq00) have an unusual frequency distribution: for most respondents the qualifications are sparse, since the majority of people don't do new courses between the ages of 33 and 42. But those who weren't present at NCDS5 have the full distribution of qualifications one would expect in any normal population, as they are reporting everything since age 16.

#### **(b) NCDS7 derived variables**

The principle here was the same as in the case of NCDS6, except that three separate sweeps needed to be compared in order to get a comprehensive longitudinal perspective. A set of four derived variables was first produced summarising highest qualification stated at the NCDS7 survey (which would be just new qualifications in the last four years, except for those not present at NCDS6) and compared to the respective level derived for NCDS6 and for NCDS5. The SPSS syntax to do this is available for download from the CLS website and from the UK Data Archive.

### **(c) NCDS8 derived variables**

Now four separate sweeps needed to be compared. A set of four derived variables was first produced summarising highest qualification stated at the NCDS8 survey (which would be just new qualifications in the last four years, except for those not present at NCDS7) and compared to the respective level derived for NCDS7, NCDS6 and NCDS5. The SPSS syntax to do this is available for download from the CLS website and from the UK Data Archive.

## **4. Longitudinal tracking of highest qualification level over time – BCS70**

### **(a) BCS70 age 30 derived variables**

The 1999/2000 sweep for BCS70 was in the field at the same time as the NCDS 1999/2000 sweep and used essentially the same questionnaire. Nevertheless, there is a fundamental difference in the wording of the educational questions: whereas NCDS members were just asked for courses and qualifications since they were last interviewed, BCS70 members were asked to summarise all their courses and qualifications since school.

The age 26 BCS70 sweep had also asked for qualifications since school, but the coverage for this postal survey had been disappointing (9,003 respondents, as opposed to 11,261 in 1999/2000). It was also found that, in the absence of an interviewer to clarify the questions, a significant number of respondents put down a level of highest qualification *above* that stated four years later in 1999/2000. This was investigated and found to be largely due to ambiguity about the status of a ‘diploma.’ The option to tick ‘diploma’ in the ‘academic qualifications’ section of the 1996 postal questionnaire was intended to pick up respondents with a teaching diploma. But numerous respondents with diplomas in vocational trades such as plumbing, carpentry and electrical engineering also ticked the box, distorting the results. So we do not use the 1996 educational information at all.

This means we produced just four derived educational variables for the BCS70 1999/2000 sweep, as opposed to the eight produced for NCDS 1999/2000 (see table in Appendix 1). The SPSS syntax used is available for download from the CLS website and from the UK Data Archive.

### **(b) BCS70 age 34 derived variables**

Using the same principle of step-by-step longitudinal accumulation of information applied with NCDS, a set of four derived variables was first produced summarising highest qualification stated at the 2004 survey (which would be just new qualifications in the last four years, except for those not present at 2000) and compared to the respective level derived at age 30. The SPSS syntax to do this is available for download from the CLS website and from the UK Data Archive.

### **(c) BCS70 age 38 derived variables**

Now three separate sweeps needed to be compared. A set of four derived variables was first produced summarising highest qualification stated at the 2008 survey (which would be just new qualifications in the last four years, except for those not present in 2004) and compared to the respective level derived at the 2000 and 2004 surveys. The SPSS syntax to do this is available for download from the CLS website and from the UK Data Archive.

## Appendix 1 Derived variable names and descriptions

### NCDS

Cohort sweep	Sweep year	Variable Name	Variable description
NCDS6	2000	hqaca00	Highest Academic level up to 2000
		hqanvq00	Highest NVQ level (academic) up to 2000
		hqvoc00	Highest NVQ level (vocational) up to 2000
		hqnvq00	Highest NVQ level (academic or vocational) up to 2000
		hiaca00	Highest Academic level at 2000 sweep
		nvqaca00	Highest NVQ level (academic) at 2000 sweep
		hivoc00	Highest NVQ level (vocational) at 2000 sweep
		hinvq00	Highest NVQ level (academic or vocational) at 2000 sweep
NCDS7	2004	nd7hachq	Highest Academic level up to 2004
		nd7hanvq	Highest NVQ level (academic) up to 2004
		nd7hvnvq	Highest NVQ level (vocational) up to 2004
		nd7hnvq	Highest NVQ level (academic or vocational) up to 2004
		nd7achq1	Highest Academic level at 2004 sweep
		nd7anvq1	Highest NVQ level (academic) at 2004 sweep
		nd7vnvq1	Highest NVQ level (vocational) at 2004 sweep
		nd7nvq1	Highest NVQ level (academic or vocational) at 2004 sweep
NCDS8	2008	nd8hachq	Highest Academic level up to 2008
		nd8hanvq	Highest NVQ level (academic) up to 2008
		nd8hvnvq	Highest NVQ level (vocational) up to 2008
		nd8hnvq	Highest NVQ level (academic or vocational) up to 2008
		nd8achq1	Highest Academic level at 2008 sweep
		nd8anvq1	Highest NVQ level (academic) at 2008 sweep
		nd8vnvq1	Highest NVQ level (vocational) at 2008 sweep
		nd8nvq1	Highest NVQ level (academic or vocational) at 2008 sweep

## Appendix 1 Derived variable names and descriptions (cont.)

### BCS70

Cohort sweep	Sweep year	Variable Name	Variable description
BCS70 age 30	2000	hiaca00	Highest Academic level at 2000 sweep
		nvqaca00	Highest NVQ level (academic) at 2000 sweep
		hivoc00	Highest NVQ level (vocational) at 2000 sweep
		hinvq00	Highest NVQ level (academic or vocational) at 2000 sweep
BCS70 age 34	2004	bd7hachq	Highest Academic level up to 2004
		bd7hanvq	Highest NVQ level (academic) up to 2004
		bd7hvnvq	Highest NVQ level (vocational) up to 2004
		bd7hnvq	Highest NVQ level (academic or vocational) up to 2004
		bd7achq1	Highest Academic level at 2004 sweep
		bd7anvq1	Highest NVQ level (academic) at 2004 sweep
		bd7vnvq1	Highest NVQ level (vocational) at 2004 sweep
		bd7nvq1	Highest NVQ level (academic or vocational) at 2004 sweep
BCS70 age 38	2008	bd8hachq	Highest Academic level up to 2008
		bd8hanvq	Highest NVQ level (academic) up to 2008
		bd8hvnvq	Highest NVQ level (vocational) up to 2008
		bd8hnvq	Highest NVQ level (academic or vocational) up to 2008
		bd8achq1	Highest Academic level at 2008 sweep
		bd8anvq1	Highest NVQ level (academic) at 2008 sweep
		bd8vnvq1	Highest NVQ level (vocational) at 2008 sweep
		bd8nvq1	Highest NVQ level (academic or vocational) at 2008 sweep



## Appendix 2 Syntax to derive to derive Highest Educational Qualification in NCDS4 (age 23 survey 1981 – variables already deposited with NCDS4 dataset)

With acknowledgements to Dougal Hutchison

\* Comment The questions asked about (first/second/third) training courses and highest qualifications achieved etc, are coded 1-31 in variables n4465 to n4679.

\* But there was some doubt about whether those originally coded 24 or 25 ('Professional Level 1' or 'Nursing Level 1' were classified correctly.

\* An alternative classification had been derived just for those cases, and stored in variables n6171 to n6376. The code below recodes the series of variables n4465 to n4679 to the revised values.

```
Do repeat xqual=n4465,n4478,n4523,n4529,n4573,n4622,n4634,n4679/  
yqual=n6171,n6176,n6216,n6221,n6226,n6231,n6236,n6376.  
if ((xqual eq 24 or 25) and (yqual ne 0))xqual=yqual.  
End repeat.
```

\* Now construct temporary variables prior to recoding into a proper hierarchical order of qualifications.

\* n4448= Quals resulting from apprenticeship/training.

```
Compute temn4448=n4448.
```

\* n4465= Quals resulting from first training course.

```
Compute temn4465=n4465.
```

\* n4478= Quals resulting from second training course.

```
Compute temn4478=n4478.
```

\* n4523= Quals resulting from third training course.

```
Compute temn4523=n4523.
```

\* n4529= Highest qual obtained since leaving school.

```
Compute temn4529=n4529.
```

Recode temn4448 to temn4529

```
(2=101)(3=102)(1=103)(24=104)(14=105)(9=106)(10=107)(5=108)(11=109)(15=110)  
(19=111)(12=112)(13=113)(16=114)(20=115)(22=116)(25=117)(17=118)(18=119)  
(21=120)(23=121)(26=122)(27=123)(28=124)(29=126)(30=127)(32=128)(31=129)  
(88=188)(else=100).
```

```
do repeat x=temn4448 temn4465 temn4478 temn4523 temn4529.
```

```
compute x=x-100.
```

```
end repeat.
```

\* Comment the above code establishes a hierarchical order of quals.

if (temn4529 eq 23 and (n6218 eq 1 or 7))temn4529=25.

\* comment now sort out the highest qual obtained.

compute maxqual2=0.

if (temn4448 le 32 and temn4448 ge 1) maxqual2=temn4448.

if (temn4465 le 32 and temn4465 ge maxqual2) maxqual2=temn4465.

if (temn4478 le 32 and temn4478 ge maxqual2) maxqual2=temn4478.

if (temn4523 le 32 and temn4523 ge maxqual2) maxqual2=temn4523.

if (temn4529 le 32 and temn4529 ge maxqual2) maxqual2=temn4529.

var labels maxqual2 'highest qual-revised for newghsq'.

compute newghsq=0.

if (maxqual2 lt 1) newghsq=16.

\* comment Note that maxqual2 doesn't take account of school exams such as O/A levels

if (maxqual2 eq 1) newghsq=15.

\* comment no code 14 - foreign qualifications.

if (n4439 eq 1) newghsq=13.

\* comment has successfully completed an apprenticeship even if got no quals on it.

\* very few code 12 as cse's not on file yet.

if (maxqual2 eq 2) newghsq=12.

if (maxqual2 eq 3 or maxqual2 eq 4 or maxqual2 eq 5 or maxqual2 eq 9 or maxqual2 eq 12 or maxqual2 eq 13) newghsq=11.

\* At this point include school quals – n4656=no of O levels, n4658=no of A levels.

if (((maxqual2 eq 0 or maxqual2 eq 1 or maxqual2 eq 2) and ((n4656 gt 0 and n4656 lt 5) or (n4655 eq 1 and (n4656 lt 1 or n4656 gt 9)))) or (maxqual2 eq 6 or maxqual2 eq 7 or maxqual2 eq 8)) newghsq=10.

if ((maxqual2 eq 3 or maxqual2 eq 4 or maxqual2 eq 5 or maxqual2 eq 9 or maxqual2 eq 12 or maxqual2 eq 13) and ((n4656 gt 0 and n4656 lt 5) or (n4655 eq 1 and (n4656 lt 1 or n4656 gt 9)))) newghsq=9.

if ((maxqual2 lt 14 and (n4656 gt 4 and n4656 lt 10)) or (maxqual2 eq 10 or maxqual2 eq 11)) newghsq=8.

if ((maxqual2 lt 17 and n4658 eq 1) or ((maxqual2 eq 14 or maxqual2 eq 15 or maxqual2 eq 16) and (n4658 lt 2)) or (maxqual2 lt 17 and (n4657 eq 1 and (n4658 lt 1 or n4658 gt 9)))) newghsq=7.

if (maxqual2 lt 17 and n4658 gt 1 and n4658 lt 10) newghsq=6.

if (maxqual2 eq 17) newghsq=5.

if (maxqual2 gt 17 and maxqual2 lt 25) newghsq=4.

if (maxqual2 eq 25) newghsq=3.

if (maxqual2 eq 26 or maxqual2 eq 27 or maxqual2 eq 28) newghsq=2.

if (maxqual2 eq 29) newghsq=1.

missing values newghsq(0).

var labels newghsq '1981 ghs qualif classification-revised'.

value labels newghsq

1 'higher degree'

2 'degree'

3 'teachers'  
 4 'Htech NCD etc'  
 5 'nurses'  
 6 '2+As'  
 7 'A or ONC or TEC'  
 8 '5+O's or craft'  
 9 'Os+'  
 10 'Os only'  
 11 'No Os –something'  
 12 'CSEs'  
 13 'Apprenticeship'  
 14 'Foreign'  
 15 'Other'  
 16 'none'.

```
Do repeat xqual=n4465,n4478,n4523,n4529,n4573,n4622,n4634,n4679/
yqual=n6171,n6176,n6216,n6221,n6226,n6231,n6236,n6376.
if ((xqual eq 24 or xqual eq 25) and (yqual ne 0))xqual=yqual.
end repeat.
```

\* comment the above code sorts out professional quals.

```
compute n4448rg=n4448.
compute n4465rg=n4465.
compute n4478rg=n4478.
compute n4523rg=n4523.
compute educqul=n4529.
```

```
recode n4448rg n4465rg n4478rg n4523rg educqul
(2=101)(3=102)(1=103)(24=104)(14=105)(9=106)(10=107)
(5=108)(4=109)(7=110)(11=111)(15=112)(19=113)(12=114)(13=115)(6=116)(8=117)
(16=118)(20=119)(22=120)(25=121)(17=122)(18=123)(21=124)(23=125)(26=126)
(27=127)(28=128)(29=130)(30=131)(32=132)(31=133)(else=100).
```

```
Do repeat x=n4448rg n4465rg n4478rg n4523rg educqul.
compute x=x-100.
end repeat.
```

\* comment the above code establishes a hierarchical order of quals.

```
if (educqul eq 27 and (n6218 eq 1 or n6218 eq 7))educqul=29.
```

\* comment now sort out the highest qual obtained.

```
compute trainqul=0.
if (n4448rg le 32 and n4448rg ge 1) trainqul=n4448rg.
if (n4465rg le 32 and n4465rg ge trainqul ) trainqul=n4465rg.
if (n4478rg le 32 and n4478rg ge trainqul ) trainqul=n4478rg.
if (n4523rg le 32 and n4523rg ge trainqul ) trainqul=n4523rg.
```

```
var labels trainqul 'highest qual in appren or training'.
```

```
var labels educqul 'highest qual on post-school educ course'.
```

```
var labels n4448rg 'highest app qual reordered'.
```

var labels n4465rg '1st training course qual reordered'.

var labels n4478rg '2nd training course qual reordered'.

var labels n4523rg '3rd training course qual reordered'.

value labels n4448rg n4465rg n4478rg n4523rg educqul

0 'none'  
1 'other'  
2 'cse'  
3 'other tech-bus'  
4 'prof lev 1'  
5 'c&g operative'  
6 'SLC'  
7 'SUPE'  
8 'CEE'  
9 'O lev'  
10 'SCE O'  
11 'RSA 1'  
12 'C&G Craft'  
13 'JIB Craft'  
14 'RSA 2'  
15 'RSA 3'  
16 'A lev'  
17 'SCE A'  
18 'C&G Adv'  
19 'ONC-OND'  
20 'TEC-BEC cert'  
21 'nursing'  
22 'C&G full tech'  
23 'C&G insig'  
24 'HNC-HND'  
25 'TEC-BEC higher'  
26 'Non-CNAA dip'  
27 'CNAA-uni dip'  
28 'prof lev 2'  
29 'non-grad teacher'  
30 '1st degree'  
31 'post-grad dip'  
32 'prof lev 3'  
33 'higher degree'.

compute trainqrg=trainqul.

compute educqrg=educqul.

recode trainqrg,educqrg(2,3,4,5=2)(6 thru 10=3)(11,14,15=4)(12,13=5)(16,17=6)  
(18,19,20=7)(21=8)(22 thru 25=9)(26,27=10)(28=11)(29=12)(30=13)(31,32,33=14).

var labels trainqrg 'Highest qual in app or training regrouped'.

var labels educqrg 'Highest qual post-school ed course regrouped'.

value labels trainqrg educqrg

0 'none'

- 1 'other'
- 2 'tech or bus etc'
- 3 'O lev equiv'
- 4 'RSA'
- 5 'Craft'
- 6 'A lev equiv'
- 7 'Technical'
- 8 'Nursing'
- 9 'Higher tech'
- 10 'Non-grad dip'
- 11 'Prof lev 2'
- 12 'Non-grad teacher'
- 13 '1st degree'
- 14 'Post-grad'.

## Appendix 3 Syntax to derive Highest Educational Qualification in NCDS6 (age 42 survey 1999/2000), and combine with information from NCDS5 (age 33 survey 1991)

With acknowledgements to Gerry Makepeace and Samantha Parsons

\*\*\*\*\* DESCRIPTION OF PROGRAM \*\*\*\*\*

- \* This program derives academic and vocational qualifications using the 2000 data only.
- \* It defines a set of binary variables showing whether an individual has a particular qualification.
- \* These variables could then be used to devise various typologies for training .
- \* We also derive the highest qualification below.
- \* All variables based on data from the 2000 surveys end in 00.

\*\*\*\*\* DEFINITION OF VARIABLES \*\*\*\*\*

- \* There are 2 important initial questions, then a series of further questions depending on the answers.
- \* The first (EDQUALS2) asks 'Have you obtained any of the qualifications on this card since xx'.
- \* If the respondent says they have some qualifications, the second question (EDQTYPxx) asks 'Which qualifications have you obtained? (GCSEs, O levels, CSE etc).
- \* The next series of questions asks, for each qualification stated, 'How many did you get at Grades (say) A-C, and 'How many at grades (say) D-G?'
- \* eg 'How many GCE O-levels did you get ..?' 'How many City and Guilds quals have you obtained ..'.
- \* Where possible, we have defined the variables using the responses to the questions asking how many qualifications of a particular type the respondent has.

\* But sometimes the respondent says 'none' or 'don't know'. This could be interpreted as follows: the respondent said 'I have the qualification' when asked the first question but on reflection realised that they did not have it. Or it could be interpreted that they remembered they got the qualification, but could not remember what grades they got.

\* In this syntax, we have assumed the second interpretation to be true, and (in the absence of any other information), attributed to them the minimum possible to be consistent with their first answer: e.g. if they say they have CSEs but don't say how many at grades 1, nor how many at grades 2-5, we assume they got just one CSE at grade 2-5.

\* 0 can be a genuine response for each of these 'how many' questions which distinguish different levels of grades. The feeder question for certain qualifications refers to all grades, while the 'how many' questions refer to different ranges of grades. Consider somebody who has 8 GCSEs all at grade A. They will say they have a gcse in response to the feeder question, but then answer 8 for the number of grade A-Cs (edgcse1) and 0 to for the number of grade D-Es (edgcse2) .

- \* The code below is easily amended to reflect the programmer's preferences.
- \* Typically the response 'I have none of those qualification' (number equals 0) and the missing values for number of qualifications (8,9,98,99) have to be changed.
- \* We have interpreted 8 and 98 as 'Do not know' and 9 and 99 as 'Not Applicable'.
- \* We have set 8 and 98 equal to 1 (ie denoting possession of the qualification) suggesting that these values reflect an inability to remember the exact number of qualifications obtained.
- \* We have set 9 and 99 equal to 0 suggesting that the respondent has considered the case further and decided that he/she does not have the qualification.
- \* The missing value dummies set 9 for edquals2 and vocqual equal to 0.

\* Not Applicable is an odd response but it seems very close to saying 'No' I do not have any of the quals.

\*\*\*\*\* INTERPRETATION OF VARIABLES \*\*\*\*\*

\* NCDS has regularly collected information on qualifications. The questions in 2000 asked 'whether the qualification has been obtained since March 1991' for individuals in NCDS V and 'whether the qualification has been obtained since March 1974' for individuals not in NCDS V.

\*\*\*\*\* EXAMPLE AND AN IMPLICATION OF OUR APPROACH \*\*\*\*\*

\* Consider the following code for the variable odeg00=1 if he/she has any other degree level qualifications .

\* We generate this from the answer to 'how many other degree level qualifications do you have?' (numothdg) using the following code.

\* recode numothdg (missing=0) (0=0) (98=1)(99=0) (1 thru hi=1) into odeg00 .

\* variable labels odeg00 "Other degree level quals such as graduate membership of professional institutes".

\* A crosstabs of numothdg by odeg00 shows that there 99 zeros for numothdeg that give odeg00=0 and 1 value of 98 that gives odeg00=1.

\* We could have defined the alternative measure, xodeg, based on the question 'Do you have any of the following qualifications?' If the individual has 'other degree level qualifications', these are recorded as the value 9 in one of the variables edqtyp14 to edqtyp21. The code for this variable is below.

\* compute xodeg=0 .

\* if (any(9,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21) ) xodeg=1 .

\* A crosstabs of xodeg by odeg00 shows that there are 99 respondents with xodeg not equal to odeg00 corresponding to the 99 zeros for numothdeg . This is a large number since there are 963 observations with odeg00=1.

\* In this derivation we use the definition similar to xodeg; this can easily be amended if one prefers to take those saying 'none' or 'don't know' to all the 'How many at grade x' questions as meaning that, after reconsideration, they didn't have that qualification after all.

\*\*\*\*\* MISSING VALUES \*\*\*\*\*

\* We begin by giving missing values the value 0. We later adjust the data for missing values.

\* Some analysis below shows that there are few genuinely missing values. The feed questions are 'Have you obtained any of the qualifications on this card?' with separate cards for educational and vocational questions. The answers to these questions are contained in edquals2 and vocqual.

\* (Be careful as there are variables with similar names for the proxy returns.) .

\* The possible answers to the feed questions are Yes, No, 8, 9, missing.

\* We treat an answer of Yes or No as saying that we have information on whether the interviewee has one or more of the qualifications.

\* The same 66 observations have missing values for both the educational and vocational questions.

\* A further 50 of the educational and 44 of the vocational answers are 8 or 9.

\* There are 28 values of 8 or 9 for both the educational and vocational questions.

\* The variables allmiss, acamiss and vocmiss capture the missing answers.

- \* No respondents have 8 or 9 for the feeder question but subsequently report a qualification.
- \* (Based on edqtyp14 and voctyp12 only).

\*\*\*\*\* PROXY INTERVIEWS \*\*\*\*\*

- \* There were 10 proxy interviews.
- \* Proxy returns occur when someone completed the form on behalf of the CM.
- \* The variable identifying a proxy interview is IntWho in the Household Grid and Housing but we did not extract that information and it is not used below.
- \* A measure (proxy=1) of whether the form is completed by a proxy is computed below .
- \* Be careful because many of the variables only refer to proxy interview eg agelfte2, edquals and vocquals are for proxies, agelfte2, edquals2 and vocqual are for interviews completed by CMs.
- \* edqtyp01 to edqtyp13, voctyp01 to voctyp11 are completed by proxies.
- \* edqtyp14 to edqtyp21, voctyp12 to voctyp17 are completed by CMs .

\*\*\*\*\* ADAPTING THE CODE \*\*\*\*\*

- \* The code be could be simplified by merging the recode and other commands. For example, the opening code below could be written as .
- \* recode edgcse1 edgcse2 (missing=0) (0=0) (98=1) (99=0) (1 thru hi =1) into gcsehi00 gcse00 .
- \* variable labels csehi00 "Has CM obtained one or more CSEs at grade 1?"
- \* / cse00 "Has CM obtained one or more CSEs at grades 2 to 5?".
- \* (Be careful. Some variables use 9 as a missing value but 9 is a legitimate value for other variables.) .
- \* The program is divided into sections because the PC on which it was developed could not handle the whole lot in one go.

\*\*\*\*\* PROGRAM STARTS \*\*\*\*\*

- \* We need data from both the NCDS5 and NCDS6 sweeps, so start by reading in both these datasets and merging them into one big file.
- \* Note you should ensure both these datafiles are sorted by NCDS case identifier (ncdsid) in advance, otherwise the 'merge' will fail.
- \* Obviously you may need to change 'd:\' in the code below to 'c:\' depending on which drive or folder contains the NCDS6 and NCDS5 data files on your computer. The 'cmi' in the NCDS5 file name stands for 'cohort member interview', to distinguish it from the mother interview and child tests etc.

```
MATCH FILES /FILE='d:\ncds6.sav'
/FILE='d:\ncds5cmi.sav'
/BY ncdsid.
EXECUTE.
```

- \* In code below, whilst constructing binary variables gcsehi00, gcse00, olevhi00, olev00 etc. which log all the different hierarchical academic qualifications achieved and at what level, we also construct a parallel series of variables xgcse, xolev, xcse, xAlev, xASlev, etc., which cater for the possibility that respondent told us they have gcse, O-levels, etc (from edqtypxx questions), but didn't tell us how many at which grades etc.

\*\*\*\*\* GCSE, O-LEVELS, CSE \*\*\*\*\*



```
recode edgcse1 (missing=0) (0=0) (98=1) (99=0) (1 thru hi =1) into gcsehi00 .
recode edgcse2 (missing=0) (0=0) (98=1) (99=0) (1 thru hi =1) into gcselo00 .
```

\* Make sure 98, 99 are before (1 thru hi) if you want them to be 0 or else they will be coded 1.  
\* In these cases, hi is 13 and 9 respectively.

```
variable labels gcsehi00 "Has CM obtained one or more GCSEs at grades A to C ?".
variable labels gcselo00 "Has CM obtained one or more GCSEs at grades D to E ?".
```

```
recode edolev1 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into olevhi00 .
recode edolev2 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into olevlo00 .
* Hi is 15 and 11.
```

```
variable labels olevhi00 "Has CM obtained one or more O levels at grades A to C?".
variable labels olevlo00 "Has CM obtained one or more O levels at grades D to E?".
```

```
recode edcse1 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into csehi00 .
recode edcse2 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into cselo00 .
* Hi is 12 and 11.
```

\* There were just two cases interviewed by proxy who had any qualifications. Their quals are recorded in EDQTYP01 instead of EDQTYP14.

\* Both had CSEs, but neither stated what grade. Assume (lower) grades 2-5.  
if (edqtyp01=3)cselo00=1.

```
variable labels csehi00 "Has CM obtained one or more CSEs at grade 1?".
variable labels cselo00 "Has CM obtained one or more CSEs at grades 2 to 5?".
```

```
compute xgcse = 0.
compute xOlev = 0.
compute xcse = 0.
```

```
if (any(1,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xgcse=1 .
```

```
if (any(2,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xOlev=1 .
```

```
if (any(3,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xcse=1 .
```

```
variable label xgcse "1 if CM has GCSE using edqtyp variables in 2000".
variable label xOlev "1 if CM has O-level using edqtyp variables in 2000".
variable label xcse "1 if CM has CSE using edqtyp variables in 2000".
```

\*\*\*\*\* AS AND A-LEVELS \*\*\*\*\*

```
recode edasl1 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into aslhi00 .
recode edasl2 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into asllo00 .
* Hi is 9 and 7.
```

```
variable labels aslhi00 "Has CM obtained one or more AS levels at grades A to C?".
variable labels asllo00 "Has CM obtained one or more AS levels at grades D to G?".
```

```
compute Naslev00=edasl1 .
recode Naslev00 (missing=0) (98=1) (99=0) .
```

variable labels Naslev00 "Number of AS level passes at grade A-C".

```
recode edgcasl1 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into alevhi00 .
recode edgcasl2 (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into alevlo00 .
```

```
compute alev00=alevhi00 + alevlo00 .
recode alev00 (0=0) (1,2=1) .
```

variable labels alevhi00 "Has CM obtained one or more A levels at grades A to C?".  
variable labels alevlo00 "Has CM obtained one or more A levels at grades D to E?".  
variable labels alev00 "Has CM obtained any A level passes at any grade?".

```
compute t1=edgcasl1 .
compute t2=edgcasl2 .
recode t1 (missing=0) (98=1) (99=0) .
recode t2 (missing=0) (98=1) (99=0) .
```

\* Someone with a 98 or 99 only gets 1 A-level.

```
compute Nalev00 = t1+t2 .
```

variable labels Nalev00 "Number of A level passes at any grade".

```
compute xASlev =0.
compute xAlev =0.
```

```
if (any(5,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xAlev=1 .
```

```
if (any(4,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xASlev=1 .
```

variable label xASlev "1 if CM has AS-level using edqtyp variables in 2000".  
variable label xAlev "1 if CM has A-level using edqtyp variables in 2000".

```
***** SCOTTISH SCHOOL QUALIFICATIONS *****
*freq edscot1 edscot2 edscot3.
```

\* The Scottish qualifications were originally in edscot1 to edscot6, but edscot4-6 only had missing values.

```
compute scotlo00=0 .
compute scot200=0 .
compute scot300=0 .
compute scot400=0 .
compute scot500=0 .
compute scot600=0 .
```

variable labels scotlo00 "Has CM obtained Scottish SCE standard grades 4-5 or equivalent?".  
variable labels scot200 "Has CM obtained Scottish SCE standard grades 1-3 or equivalent?".  
variable labels scot300 "Has CM obtained Scottish SUPE-SLC lower or ordinary grade?".  
variable labels scot400 "Has CM obtained Scottish SCE-SUPE-SLC higher grade or equivalent?".  
variable labels scot500 "Has CM obtained Scottish Certificate of 6th Year Studies?".  
variable labels scot600 "Has CM obtained other Scottish school qualification?".

\* Scottish higher equivalent to A level in IFS definitions.

```

if ( any(1,edscot1,edscot2,edscot3) ) scotlo00=1 .
if ( any(2,edscot1,edscot2,edscot3) ) scot200=1 .
if ( any(3,edscot1,edscot2,edscot3) ) scot300=1 .
if ( any(4,edscot1,edscot2,edscot3) ) scot400=1 .
if ( any(5,edscot1,edscot2,edscot3) ) scot500=1 .
if ( any(6,edscot1,edscot2,edscot3) ) scot600=1 .

***** DEGREES *****
* Value 9 for numdeg indicates 'not answered', so we assume no degree (only 1 such case anyway).
recode numdeg (missing=0) (0=0) (9=0) (1 thru hi=1) into deg00 .
* Hi is 4.

variable labels deg00 "Has CM obtained a degree?".

recode numothdg (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into odeg00 .
* hi = 6.
variable labels odeg00 "Other degree level quals such as graduate membership of professional
institutes".

compute xdeg=0 .
if (any(8,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21) ) xdeg=1 .

compute xodeg=0 .
if (any(9,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21) ) xodeg=1 .

compute xhdeg=0 .
if (any(10,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21) ) xhdeg=1
.

variable labels xdeg "Degree using edqtyp variables".
variable labels xodeg "Other degree level qualifications using edqtyp variables".
variable labels xhdeg "Higher degree using edqtyp variables".

recode numhghdg (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into hdeg00 .
*hi = 3, no 98 or 99.
variable labels hdeg00 "Has CM obtained a higher degree?".

***** NURSING, DIPLOMAS, PGCE *****

*There are 3 values for edqtyp21; variables edqtyp22 to edqtyp26 were originally present, but all values
were missing.

compute dip00=0 .
if (any(7,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)
or (numotht > 0 & numotht < 15)) dip00=1 .

variable labels dip00 "Diploma-Certificate-Teacher Training Qual (not pgce) in Higher Ed".
* 357 have 1 teacher training qualification. 1 person has 12.

recode numparam (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into nurse00 .
*hi=15 No 98 or 99.

variable labels nurse00 "Has CM obtained any nursing or paramedic qualifications?".

```

```
compute xoteach=0
if (any(13,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xoteach=1
```

```
compute xnurse=0
if (any(11,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,edqtyp20,edqtyp21)) xnurse=1
```

```
variable labels xnurse "Any nursing or paramedic qualifications using edqtyp variables".
variable labels xoteach "Other teaching qualification (not PGCE) using edqtyp variables".
```

```
compute pgce00=0
if (any(12,edqtyp14,edqtyp15,edqtyp16,edqtyp17,edqtyp18,edqtyp19,
edqtyp20,edqtyp21)) pgce00=1
```

```
variable labels pgce00 "Has CM obtained a PGCE or other postgraduate certificate in HE?".
```

```
execute .
*****
***** FIRST BLOCK FOR RUNNING PROGRAM *****
*****
```

```
***** ONC or OND AND HNC or HND *****
recode voconc (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into oncond00
* hi=7 .
```

```
variable labels oncond00 "Does CM have ONC or OND?".
```

```
recode vohnc (missing=0) (0=0) (9=0) (98=1) (99=0) (1 thru hi=1) into hnchnd00
* hi=4. One value of 9.
```

```
variable labels hnchnd00 "Does CM have HNC or HND?".
```

```
***** BTECS *****
```

```
compute btec100=0
compute btec200=0
```

```
variable labels btec100 "Does CM have a BTEC etc First-General Certificate?".
variable labels btec200 "Does CM have a BTEC etc First-General Diploma?".
```

```
compute btec300=0
compute btec400=0
```

```
variable labels btec300 "Does CM have a BTEC etc National Certificate Diploma?".
variable labels btec400 "Does CM have a BTEC etc Higher Certificate Diploma?".
```

```
compute btec500=0
```

```
variable labels btec500 "Does CM have other BTEC qualification?".
```

```
if (any(1,btecle1,btecle2,btecle3,btecle4,btecle5,btecle6,btecle7,btecle8)) btec100=1
```

```
if (any(2,btecle1,btecle2,btecle3,btecle4,btecle5,btecle6,btecle7,btecle8)) btec200=1
```

if (any(3,bteclelev,bteclelev2,bteclelev3,bteclelev4,bteclelev5,bteclelev6,bteclelev7,bteclelev8)) btec300=1 .

if (any(4,bteclelev,bteclelev2,bteclelev3,bteclelev4,bteclelev5,bteclelev6,bteclelev7,bteclelev8)) btec400=1 .

if (any(5,bteclelev,bteclelev2,bteclelev3,bteclelev4,bteclelev5,bteclelev6,bteclelev7,bteclelev8)) btec500=1 .

\* Variables bteclelev9 to bteclelev15 were originally present, but all values missing. All have first or general certificate.

\* Some values equal to 9 eg 89 observations with 9 for bteclelev.

\*\*\*\*\* CITY AND GUILDS \*\*\*\*\*

compute city100=0 .

compute city200=0 .

compute city300=0 .

compute city400=0 .

compute city500=0 .

variable labels city100 "Does CM have a City and Guilds Part 1 " .

variable labels city200 "Does CM have a City and Guilds Part2 or Craft or Intermediate" .

variable labels city300 "Does CM have a City and Guilds Part3 or Final or Advanced Craft" .

variable labels city400 "Does CM have a City and Guilds Part4 or Career Extension or Full technological" .

variable labels city500 "Does CM have a City and Guilds Other C&G qualification" .

If (any(1,citylev,citylev2,citylev3,citylev4,citylev5,citylev6,citylev7,citylev8,citylev9,citylev10,citylev11,citylev12,citylev13,citylev14,citylev15)) city100=1 .

If ( any(2,citylev,citylev2,citylev3,citylev4,citylev5,citylev6,citylev7,citylev8,citylev9,citylev10,citylev11,citylev12,citylev13,citylev14,citylev15)) city200=1 .

If any(3,citylev,citylev2,citylev3,citylev4,citylev5,citylev6,citylev7,citylev8,citylev9,citylev10,citylev11,citylev12,citylev13,citylev14,citylev15) city300=1 .

If any(4,citylev,citylev2,citylev3,citylev4,citylev5,citylev6,citylev7,citylev8,citylev9,citylev10,citylev11,citylev12,citylev13,citylev14,citylev15) city400=1 .

If any(5,citylev,citylev2,citylev3,citylev4,citylev5,citylev6,citylev7,citylev8,citylev9,citylev10,citylev11,citylev12,citylev13,citylev14,citylev15) city500=1 .

\*\*\*\* RSA \*\*\*\*\*

compute rsa100=0 .

compute rsa200=0 .

compute rsa300=0 .

compute rsa400=0 .

compute rsa500=0 .

variable labels rsa100 "Does CM have RSA certificate?".

variable labels rsa200 "Does CM have RSA First Diploma?".  
variable labels rsa300 "Does CM have RSA Advanced Diploma or Certificate?".  
variable labels rsa400 "Does CM have RSA Higher Diploma" .  
variable labels rsa500 "Does CM have other RSA qualification?".

```
If (any(1,rsalev,rsalev2,rsalev3,rsalev4,rsalev5,rsalev6,rsalev7,rsalev8,rsalev9,rsalev10,rsalev11,rsalev12)) rsa100=1 .  
If (any(2,rsalev,rsalev2,rsalev3,rsalev4,rsalev5,rsalev6,rsalev7,rsalev8,rsalev9,rsalev10,rsalev11,rsalev12)) rsa200=1 .  
If (any(3,rsalev,rsalev2,rsalev3,rsalev4,rsalev5,rsalev6,rsalev7,rsalev8,rsalev9,rsalev10,rsalev11,rsalev12)) rsa300=1 .  
If (any(4,rsalev,rsalev2,rsalev3,rsalev4,rsalev5,rsalev6,rsalev7,rsalev8,rsalev9,rsalev10,rsalev11,rsalev12)) rsa400=1 .  
If (any(5,rsalev,rsalev2,rsalev3,rsalev4,rsalev5,rsalev6,rsalev7,rsalev8,rsalev9,rsalev10,rsalev11,rsalev12)) rsa500=1 .
```

\* rsalev10 to rsalev12 only have 1 observation each. All values are for first diplomas (2).

\*\*\*\* PITMANS \*\*\*\*\*.

\* Pitmans are only in the 2000 data.

```
compute pit100=0 .  
compute pit200=0 .  
compute pit300=0 .  
compute pit400=0 .
```

variable labels pit100 "Does CM have a Pitmans level 1?".  
variable labels pit200 "Does CM have a Pitmans level 2?".  
variable labels pit300 "Does CM have a Pitmans level 3?".  
variable labels pit400 "Does CM have other Pitmans qualification?".

```
If (any(1,pitlev,pitlev2,pitlev3,pitlev4)) pit100=1 .  
If (any(2,pitlev,pitlev2,pitlev3,pitlev4)) pit200=1 .  
If (any(3,pitlev,pitlev2,pitlev3,pitlev4)) pit300=1 .  
If (any(4,pitlev,pitlev2,pitlev3,pitlev4)) pit400=1 .
```

\* pitlev5-7 were originally present, but all missing values.

\*\*\*\*\* NVQ \*\*\*\*\*.

\*nvqs (current sweep only) .

```
compute nvq100=0 .  
compute nvq200=0 .  
compute nvq300=0 .  
compute nvq400=0 .  
compute nvq500=0 .  
compute nvq600=0 .  
compute nvq700=0 .  
compute nvq800=0 .
```

variable labels nvq100 "Does CM have an NVQ at Level 1?".  
variable labels nvq200 "Does CM have an NVQ at Level 2?".  
variable labels nvq300 "Does CM have an NVQ at Level 3?".  
variable labels nvq400 "Does CM have an NVQ at Level 4?".

variable labels nvq500 "Does CM have an NVQ at Level 5?".  
variable labels nvq600 "Does CM have an NVQ at Level 6?".  
variable labels nvq700 "Does CM have Trusts towards NVQ-SVQ?".  
variable labels nvq800 "Does CM have other NVQ?".

If (any(1,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq100=1 .  
If (any(2,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq200=1 .  
If (any(3,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq300=1 .  
If (any(4,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq400=1 .  
If (any(5,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq500=1 .  
If (any(6,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq600=1 .  
If (any(7,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq700=1 .  
If (any(8,nvqlev,nvqlev2,nvqlev3,nvqlev4,nvqlev5,nvqlev6)) nvq800=1 .

\*nvqlev7 to nvqlev10 were originally present, but had no values.

\*\*\*\*\* GNVQ \*\*\*\*\*.

\* gnvqs (current sweep only)

compute gnvq100=0 .  
compute gnvq200=0 .  
compute gnvq300=0 .  
compute gnvq400=0 .

variable labels gnvq100 "Does CM have GNVQ Foundation?".  
variable labels gnvq200 "Does CM have GNVQ Intermediate?".  
variable labels gnvq300 "Does CM have GNVQ Advanced?".  
variable labels gnvq400 "Does CM have other GNVQ qualification?".

If (any(1,gnvlev,gnvlev2,gnvlev3)) gnvq100=1 .  
If (any(2,gnvlev,gnvlev2,gnvlev3)) gnvq200=1 .  
If (any(3,gnvlev,gnvlev2,gnvlev3)) gnvq300=1 .  
If (any(4,gnvlev,gnvlev2,gnvlev3)) gnvq400=1 .

\*gnvlev4 and gnvlev5 were originally present, but had no values.

\*\*\* OTHER QUALIFICATIONS \*\*\*\*\*.

recode vocappr (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into appren00 .  
\* hi is 7.

variable labels appren00 "Has CM obtained any recognised trade apprenticeship qualifications?".

recode vochgv (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into hgv00 .  
\* hi is 3.

variable labels hgv00 "Has CM obtained at least one hgv qualification?".

recode vocoth (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into othvoc00 .  
\* hi is 15.

variable labels othvoc00 "Has CM obtained at least one other vocational qualification?".

```
*****
***** MISSING VALUE ANALYSIS *****
*****
```

```
* edquals2, vocqual      1='Yes' 2='No' .
```

```
* We argue that if initial feeder questions (edquals2, vocqual) are answered 'properly' then we know whether the individual has a qualification or not. If the answer is missing or 'do not know' then we do not know. The code below sets all the educational qualifications to missing if edquals is not answered Yes or No or NA and all the vocational qualifications to missing if vocqual is not answered Yes or No or 'Not Applicable'.
```

```
recode edquals2 (2=0) (1=0) (8=1) (9=0) (missing=2) into acamiss.
```

```
* There are also 10 proxy cases who give us a definite yes or no on academic qualifications (14 say no, 2 say yes).
```

```
if (edquals=1 or edquals=2)acamiss=0.
```

```
variable label acamiss 'Dummy for missing value for do you have an educational qualification'.
```

```
value label acamiss 0 'not missing' 1 '8 (do not know)' 2 'missing'.
```

```
recode vocqual (1=0) (2=0) (8=1) (9=0) (missing=2) into vocmiss.
```

```
* vocmiss adjusted for nurse00 (nurse00 appears as an educational qualification) below.
```

```
variable label vocmiss 'Dummy for missing value for do you have a vocational qualification'.
```

```
value label vocmiss 0 'not missing' 1 '8 (do not know)' 2 'missing'.
```

```
do if (vocmiss eq 0 and acamiss eq 0) .
```

```
  compute allmiss =0 .
```

```
  else if (vocmiss eq 1 and acamiss eq 0) .
```

```
    compute allmiss =1 .
```

```
  else if (vocmiss eq 0 and acamiss eq 1) .
```

```
    compute allmiss =2 .
```

```
  else if (vocmiss eq 1 and acamiss eq 1) .
```

```
    compute allmiss =3 .
```

```
  else if (vocmiss eq 2 and acamiss eq 2) .
```

```
    compute allmiss =4 .
```

```
end if .
```

```
variable label allmiss 'Missing values, Do you have any of these qualifications?'
```

```
value label allmiss 0 'not missing' 1 '8 vocat quals, not missing for educ quals'
```

```
  2 'vocat quals not missing, 8 for educ quals' 3 '8 for both vocat and educ quals'
```

```
  4 'missing for both vocational & educational quals'.
```

```
***** Set missing values *****.
```

```
do if (acamiss gt 0) .
```

```
  compute gcsehi00 =-9 .
```

```
  compute gcselo00 =-9 .
```

```
  compute olevhi00 =-9 .
```

```
  compute olevlo00 =-9 .
```

```
  compute csehi00 =-9 .
```

```
  compute cselo00 =-9 .
```

```
  compute aslhi00 =-9 .
```

```
  compute asllo00 =-9.
```



```

compute naslev00 = -9.
compute alevhi00 = -9.
compute alevlo00 = -9.
compute alev00 = -9.
compute nalev00 = -9.
compute scotlo00 = -9.
compute scot200 = -9.
compute scot300 = -9.
compute scot400 = -9.
compute scot500 = -9.
compute scot600 = -9.
compute dip00 = -9.
compute deg00 = -9.
compute odeg00 = -9.
compute dip00 = -9.
compute hdeg00 = -9.
compute nurse00 = -9.
compute pgce00 = -9.
end if .

```

if (nurse00 eq 1) vocmiss eq 0 .

\* nurse is a vocational qualification under some typologies eg nvq but educational here.

```

do if (vocmiss gt 0) .
  compute oncond00 =-9 .
  compute hnchnd00 =-9 .
  compute btec100=-9 .
  compute btec200=-9 .
  compute btec300=-9 .
  compute btec400=-9 .
  compute btec500=-9 .
  compute city100 =-9 .
  compute city200 =-9 .
  compute city300 =-9 .
  compute city400 =-9 .
  compute city500 = -9 .
  compute rsa100 = -9 .
  compute rsa200 =-9 .
  compute rsa300 =-9 .
  compute rsa400 =-9 .
  compute rsa500 =-9 .
  compute pit100 =-9 .
  compute pit200 =-9 .
  compute pit300 =-9 .
  compute pit400 =-9 .
  compute nvq100 =-9 .
  compute nvq200 =-9 .
  compute nvq300 =-9 .
  compute nvq400 =-9 .
  compute nvq500 =-9 .
  compute nvq600 =-9 .
  compute nvq700 =-9 .
  compute nvq800 =-9 .
  compute gnvq100 =-9 .

```

```

compute gnvq200 =-9 .
compute gnvq300 =-9 .
compute gnvq400 =-9 .
compute appren00=-9 .
compute hgv00 =-9 .
compute othvoc00 =-9 .
end if .

```

missing value gcsehi00 to pgce00 oncond00 to othvoc00 (-9) .

```

compute proxy = 0 .
if (edquals gt 0) proxy =1 .
* if (vocquals gt 0) proxy1 =1 not needed as proxy = proxy1.

```

variable label proxy 'Was interview completed by a proxy'.

\* Acamiss=vocmiss=1 when proxy=1 so proxy 'accounts' for 26 of the missing values.

```

compute acamiss1=0 .
compute vocmiss1=0 .

```

```

if (missing(edqtyp14) eq 1 and missing(edqtyp15) eq 1 and missing(edqtyp16) eq 1 and
missing(edqtyp17) eq 1 and missing(edqtyp18) eq 1 and missing(edqtyp19) eq 1 and
missing(edqtyp20) eq 1 and missing(edqtyp21) eq 1) acamiss1=1 .

```

```

if (missing(voctyp12) eq 1 and missing(voctyp13) eq 1 and missing(voctyp14) eq 1 and
missing(voctyp15) eq 1 and missing(voctyp16) eq 1 and missing(voctyp17) eq 1) vocmiss1=1 .

```

\*\*\*\*\* GOVERNMENT TRAINING SCHEMES \*\*\*\*\*.

```

recode numyts (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into yts00 .
* hi is 15.

```

variable labels yts00 "Has CM done a YTS course?".

```

recode numgov (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into gov00 .
* hi is 15.

```

variable labels gov00 "Has CM done another gov course (including New Deal)?".

```

recode numap (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into modapp00 .
* hi is 3.

```

variable labels modapp00 "Has CM done a modern apprenticeship?".

execute .

```

*****
***** SECOND BLOCK WHEN RUNNING PROGRAM *****
*****

```

\*\*\*\*\* OTHER COURSES \*\*\*\*\*

```
recode numfqual (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into nfail00.  
*Hi =11.  
variable labels nfail00 "Number of courses started where CM did not obtain the qualification".
```

```
recode numac (missing=0) (0=0) (98=1) (99=0) (1 thru hi=1) into access00 .  
* hi is 10.
```

```
variable labels access00 "Has CM done an access course?".
```

```
compute workc00=numwrktr .  
recode workc00 (missing=0) (0=0) (98=1) (99=0) .  
* max is 30.
```

```
variable labels workc00 "Number of other work related courses".
```

\*\*\*\*\* NON WORK RELATED COURSES \*\*\*\*\*

```
compute leis00=numleis .  
recode leis00 (missing=0) (0=0) (98=1) (99=0) .  
* max is 15.
```

```
variable labels leis00 "Number of courses for interest or leisure".
```

\*\*\*\* READING, WRITING AND MATHS COURSES \*\*\*\*\*

```
recode numread (missing=0) (0=0) (98=1) (99=0) (1=1) (2=2) (3 thru hi=3) into read00 .  
* max is 11.
```

```
variable labels read00 "Number of courses to improve reading".
```

```
recode numwrite (missing=0) (0=0) (98=1) (99=0) (1=1) (2=2) (3 thru hi=3) into write00 .  
* max is 11.
```

```
variable labels write00 "Number of courses to improve writing".  
* creative writing ?.
```

```
recode nummaths (missing=0) (0=0) (98=1) (99=0) (1=1) (2=2) (3 thru hi=3) into maths00 .  
* max is 15.
```

```
variable labels maths00 "Number of courses to improve maths".  
value labels read00 write00 maths00 3 "3 or more" .
```

\*\*\* THERE IS A LOT ABOUT CURRENT COURSES THAT WE HAVE NOT USED \*\*\*\*\*

\*\*\*\*\*  
\*\*\*\*\* DUMMIES FOR GROUPS OF QUALIFICATIONS \*\*\*\*\*  
\*\*\*\*\*

\*\*\*\*\* HAS CM OBTAINED A PARTICULAR ACADEMIC QUALIFICATION? \*\*\*\*\*

```
do if (acamiss gt 0) .
```

```

        compute Aca000=-9 .
        compute Aca100=-9 .
        compute Aca200=-9 .
        compute Aca300=-9 .
        compute Aca400=-9 .
        compute Aca500=-9 .
        compute Aca600=-9 .
        compute Aca700=-9 .
        compute Aca800=-9 .
else .
        compute Aca000=1 .
        compute Aca100=0 .
        compute Aca200=0 .
        compute Aca300=0 .
        compute Aca400=0 .
        compute Aca500=0 .
        compute Aca600=0 .
        compute Aca700=0 .
        compute Aca800=0 .
end if .

```

missing values Aca000 to Aca800 (-9) .

\* AcaX00 dummy variables variables showing whether CM has one of a group of qualifications.

variable labels Aca000 "1 if CM has no academic qualifications".

variable labels Aca100 "1 if CM has bad O-levels " .

variable labels Aca200 "1 if CM has CSE 2-5, Other Scots " .

variable labels Aca300 "1 if CM has good O-levels " .

variable labels Aca400 "1 if CM has 1 A-level or more than 1 AS at grade A-C " .

variable labels Aca500 "1 if CM has 2 or more A-levels".

variable labels Aca600 "1 if CM has diploma " .

variable labels Aca700 "1 if CM has degree, PGCE, other degree level " .

variable labels Aca800 "1 if CM has higher degree".

value labels aca000 0 "At least one academic qualification" 1 "No academic qualifications".

\* Bad O-levels

GCSEs D to E, GCEs D to E, Scottish SCE standard grades 4-5' .

\* Good O-levels

CSE grade 1, GCSE grades A-C, GCE grades A-C, Scots standard grades 1-3, Scots lower or ordinary.

\* 2 or more A-LEVELS

2 or more A-levels, Scottish higher grade or Certificate of 6th Year Studies .

\* If we only know they have a qualification (e.g. O levels), but don't know how many at which grades (i.e. xolev=1 but olevhi00 and olevlo00 undefined), we assume the minimum: e.g. one 'bad' O level.

\* Same logic applies for all binary variables below beginning with 'x'.

```

do if (hdeg00 =1 or xhdeg=1) .
        compute Aca800=1 .
        compute Aca000=0 .
end if .

```

```

do if (any(1,deg00,odeg00,xdeg,xodeg,pgce00)) .

```

```

        compute Aca700=1 .
        compute Aca000=0 .
end if .

do if (dip00 eq 1 or xoteach=1) .
        compute Aca600=1 .
        compute Aca000=0 .
end if .

do if (Nalev00 gt 1 or scot400 eq 1 or scot500 eq 1) .
        compute Aca500=1 .
        compute Aca000=0 .
end if .

do if (Nalev00 eq 1 or Naslev00 gt 1 or xalev=1) .
        compute Aca400=1 .
        compute Aca000=0 .
end if .

do if (any(1,gcsehi00,olevhi00,csehi00,scot200,scot300)) .
        compute Aca300=1 .
        compute Aca000=0 .
end if .

```

\* If only one AS level A-C, or AS levels with lower or unspecified grades, set quals = '0 levels or good GCSEs'.

```

do if (Naslev00 eq 1 or (xaslev=1 and (edas1=0 or sysmis(edas1)))) .
        compute Aca300=1 .
        compute Aca000=0 .
end if .

do if (cselo00 eq 1 or scot600 eq 1 or xcse=1) .
        compute Aca200=1 .
        compute Aca000=0 .
end if .

do if (any(1,gcselo00,olevlo00,scotlo00, xgcse, xolev)) .
        compute Aca100=1 .
        compute Aca000=0 .
end if .

```

\*\*\*\*\* HAS CM OBTAINED A PARTICULAR VOCATIONAL QUALIFICATION? \*\*\*\*\*.

```

do if (Vocmiss gt 0) .
        compute Voc000=-9 .
        compute Voc100=-9 .
        compute Voc200=-9 .
        compute Voc300=-9 .
        compute Voc400=-9 .
        compute Voc500=-9 .
else .
        compute Voc000=1 .
        compute Voc100=0 .
        compute Voc200=0 .

```

```

        compute Voc300=0 .
        compute Voc400=0 .
        compute Voc500=0 .
end if .

```

missing values Voc000 to Voc500 (-9) .

\* VocX00 dummy variables variables showing whether CM has one of a group of qualifications.

```

variable labels Voc000 "1 if CM has no vocational qualifications".
variable labels Voc100 "1 if CM has vocational qualification equivalent to NVQ1".
variable labels Voc200 "1 if CM has vocational qualification equivalent to NVQ2".
variable labels Voc300 "1 if CM has vocational qualification equivalent to NVQ3".
variable labels Voc400 "1 if CM has vocational qualification equivalent to NVQ4".
variable labels Voc500 "1 if CM has vocational qualification equivalent to NVQ5".

```

```

value labels voc000 0 "At least one vocational qualification" 1 "No vocational qualifications".

```

```

*
    100 'C&G Opo & Other, RSA 1, HGV, Other tech or bus qualification'
    200 'C&G Craft-Inter-Ord-Part1-Cant say, RSA2, Insig,JIB NJC etc'
    300 'C&G Advanced-Final-Part II or III, ONC-OND, RSA 3 '
    400 'Nursing, BTEC HNC-HND, HNC-HND, C&G Full Tech (FTC)'
    500 'NVQ awarded at level 5'.

```

```

do if (any(1,nvq500,nvq600)) .
    compute Voc500=1 .
    compute Voc000=0 .
end if .

```

```

do if (any(1,nurse00,hnchnd00,btec400,rsa400,nvq400)) .
    compute Voc400=1 .
    compute Voc000=0 .
end if .

```

```

do if (any(1,city200,city300,city400,oncond00,btec300,rsa300,pit300,nvq300,gnvq300)) .
    compute Voc300=1 .
    compute Voc000=0 .
end if .

```

```

do if (any(1,city100,btec200,rsa200, pit200,appren00,nvq200,gnvq200)) .
    compute Voc200=1 .
    compute Voc000=0 .
end if .

```

```

do if (any(1,city500,btec100,rsa100,pit100,nvq100,gnvq100,hgv00,othvoc00,rsa500,
    btec500,pit400,nvq700,nvq800,gnvq400)) .
    compute Voc100=1 .
    compute Voc000=0 .
end if .

```

\*\*\*\*\*

\*\*\*\*\* HIGHEST QUALIFICATIONS \*\*\*\*\*  
 \*\*\*\*\*

- \* This code computes the highest academic qualification, the highest vocational qualification and the highest NVQ level recorded in the 2000 survey.
- \* Previous published work has used slightly different definitions of these variables so you may need to adapt the code below.
- \* Check bad O-levels, many descriptions give them NVQ level of 0
- \* All vocational stuff needs checking especially pitmans btec rsa .

\*\*\*\*\* HIGHEST ACADEMIC QUALIFICATION \*\*\*\*\*

```
do if (acamiss gt 0) .
  compute Hiaca00=-9 .

else if (Aca800=1) .
  compute Hiaca00=8 .

else if (Aca700=1) .
  compute Hiaca00=7 .

else if (Aca600=1) .
  compute Hiaca00=6 .

else if (Aca500=1) .
  compute Hiaca00=5 .

else if (Aca400=1) .
  compute Hiaca00=4 .

else if (Aca300=1) .
  compute Hiaca00=3 .

else if (Aca200=1) .
  compute Hiaca00=2 .

else if (Aca100=1) .
  compute Hiaca00=1 .

else if (acamiss eq 0) .
  compute Hiaca00=0 .

end if .
```

variable label Hiaca00 'Highest Academic level at 2000 sweep'.

value labels Hiaca00  
 -9 Missing  
 0 'None'  
 1 'Bad O-levels'  
 2 'CSE 2-5, other Scottish school qualification'  
 3 'Good O levels'  
 4 '1 A level or more than 1 AS level at grade A-C'  
 5 '2 or more A-levels'

6 'Diploma'  
7 'Degree, PGCE, other degree level'  
8 'Higher degree ' .

\* Notice diploma=degree .

recode Hiaca00 (-9=-9) (0=0) (1=1) (2=1)(3=2)(4=2)(5=3)(6=4)(7=4)(8=5) into nvqaca00 .

variable label nvqaca00 'Highest NVQ level (academic) at 2000 sweep'.

recode Hiaca00 (-9=-9) (0=0) (1=1) (2=1)(3=2)(4=2)(5=3)(6=4)(7=5)(8=6) into ghmaca00 .

variable label ghmaca00 'Highest academic qualification in 2000 survey ghm measure see also IFS'.

value labels ghmaca00 0 'None' 1 'Bad O levels, CSE 2-5' 2 'Good O-levels, 1 A-level'  
3 '2 or more A-levels' 4 'Sub-degree' 5 'Degree' 6 'Higher Degree' .

\*\*\*\*\* HIGHEST VOCATIONAL QUALIFICATION \*\*\*\*\*.

do if (vocmiss gt 0) .  
compute HiVoc00=-9 .

else if (Voc500 eq 1) .  
compute HiVoc00=5 .

else if (Voc400 eq 1) .  
compute HiVoc00=4 .

else if (Voc300 eq 1) .  
compute HiVoc00=3 .

else if (Voc200 eq 1) .  
compute HiVoc00=2 .

else if (Voc100 eq 1) .  
compute HiVoc00=1 .

else if (vocmiss eq 0) .  
compute HiVoc00=0 .

end if .

variable label HiVoc00 'Highest NVQ level (vocational) at 2000 sweep'.

value labels HiVoc00  
-9 Missing  
0 'None '  
1 'C&G Opo & Other, RSA 1, HGV, Other tech or bus qualification'  
2 'C&G Craft-Inter-Ord-Part1-Cant say, RSA2, Insig,JIB NJC etc'  
3 'C&G Advanced-Final-Part II or III, ONC-OND, RSA 3 '  
4 'Nursing, BTEC HNC-HND, HNC-HND, C&G Full Tech (FTC)'  
5 'NVQ awarded at level 5'.



\*\*\*\*\* Highest NVQ whatever type \*\*\*\*\*.

```
compute HiNVQ00=max(nvqaca00,HiVoc00) .
```

```
variable label HiNVQ00 'Highest NVQ level (academic or vocational) at 2000 sweep' .
```

```
missing value Hiaca00 nvqaca00 ghmaca00 HiVoc00 HiNVQ00 (-9) .
```

```
format Hiaca00 nvqaca00 ghmaca00 HiVoc00 HiNVQ00 (f8.0) .
```

```
**MERGING NCDS5 QUALIFICATION DATA WITH 2000 DATA.
```

```
**CHANGED 2000 TO MATCH NCDS33.
```

```
**simple way of recoding highest academic/vocational qualifications - to match NCDS at 33 hqual vars.
```

```
**ghmaca00 - levels of academic qualifications: ranges from 0 to 6: with 5 = degree and 6 = higher degree.
```

```
.
```

```
**hivoc00 - levels of vocational NVQ qualifications: ranges from 0 to 5: with 5 = NVQ level 5.
```

```
**variable HINVQ00 combines the information from the two variables above with a range of 0 to 5, BUT level 5
```

```
represents ghmaca00 level 6 and hivoc00 level 5 only. Surely it should be ghmaca00 level 5 AND 6 and hivoc00 level 5?.
```

```
* Note from Brian Dodgeon November 2010 in answer to above (much earlier) comment - I suspect this confusion stems from the official re-classification in the late 1990s:
```

```
* at one time a (first) degree was deemed equivalent to NVQ level 5, but this was revised to NVQ level 4 by the Qualifications and Curriculum Authority. So there was no longer a need for levels of academic quals to go up to 6.
```

```
* It seems therefore that the 'ghm' version is now obsolete, so we have not derived it for the 2004 and 2008 versions of this syntax.
```

```
**made such a variable .
```

```
compute hq2ka = ghmaca00.
```

```
if (hivoc00 > ghmaca00) hq2ka = hivoc00.
```

```
recode hq2ka (5,6=5).
```

```
** Highest qualification at NCDS5 (age 33) – NB: Uses NCDS5 variables n501513 to n501541.
```

```
do rep x=n501441 to n501469/y=n501513 to n501541.
```

```
if x=37 and range (y,1,36) x=y.
```

```
if x=37 and missing (y) or y=0 y=37.
```

```
end rep.
```

```
compute hqual33=-2.
```

```
do rep x=n501441 to n501469.
```

```
if any (x,37) hqual33=0.
```

```
if (any (x,10,25,1) and hqual33 < 1) hqual33=1.
```

```

if (any (x,20,19,18,17,13,14,12,11,7,6,4,3,2) and hqual33 < 2) hqual33=2.
if (any (x,23,21,15,09,08,05) and hqual33 < 3) hqual33=3.
if (any (x,30,29,28,27,26,24,22,16) and hqual33 < 4) hqual33=4.
if (range (x,31,33) and hqual33 < 5) hqual33=5.
end rep.
var labels hqual33 "Highest qual gained at age 33 – based on NCDS5 variables ".
value labels hqual33 0 'No qualification' 1 'CSE 2-5/equiv NVQ1' 2 'O Level/equiv NVQ2' 3 'A Level/equiv
NVQ3' 4 'Higher qual NVQ4' 5 'Degree/higher NVQ5/6' -1 'No information'.
formats hqual33 (f2.0).
recode hqual33 (-2=-1).
missing values hqual33 (-1).

```

\*\*Highest qualification variable combining information of NCDS cohort at age 33.

```

compute hq332k = hq2ka.
if (hq2ka = 0 and hqual33 >=0) hq332k = hqual33.
do if (hq2ka > 0 and hqual33 >= 0).
if (hqual33 < hq2ka) hq332k = hq2ka.
if (hqual33 = hq2ka) hq332k = hq2ka.
if (hqual33 > hq2ka) hq332k = hqual33.
end if.

```

```

variable labels hq332k 'highest NVQ qualification - academic or vocational - recoding 2000 to match
NCDS5'.
format hq332k (f3.0).
* frequencies variables=hq332k.

```

```

*****
**some qualifications coded differently at each sweep, so made individual qualification vars at age 33
to re-create various highest qualification variables at 33 to match 2000.
*****

```

```

do if (n501441 >= 0).
compute noquals = 0.
compute cse2_5 = 0.
compute cse1 = 0.
compute oleva_c = 0.
compute gcsea_c = 0.
compute alev = 0.
compute scotoa_c = 0.
compute scots1_3 = 0.
compute scothg = 0.
compute scot6th = 0.
compute rsa1 = 0.
compute rsa2 = 0.
compute rsa3 = 0.
compute cg_op = 0.
compute cg_cio1 = 0.
compute cg_afp23 = 0.
compute cg_ft = 0.
compute cg_oth = 0.
compute cg_dk = 0.
compute cg_ia = 0.
compute jibnjcct = 0.

```

```

compute onchnd = 0.
compute hnchnd = 0.
compute bt_ngcd = 0.
compute bt_hhncd = 0.
compute othtbq = 0.
compute fprofq = 0.
compute pprofq = 0.
compute nurse = 0.
compute cert = 0.
compute hdippgce = 0.
compute degree = 0.
compute pgraddip = 0.
compute hdegree = 0.
do repeat x = n501441 to n501469.
if (x = 37) noquals = 1.
if (x = 1) cse2_5 = 1.
if (x = 2) cse1 = 1.
if (x = 3) oleva_c = 1.
if (x = 4) gcsea_c = 1.
if (x = 5) alev = 1.
if (x = 6) scotoa_c = 1.
if (x = 7) scots1_3 = 1.
if (x = 8) scothg = 1.
if (x = 9) scot6th = 1.
if (x = 10) rsa1 = 1.
if (x = 11) rsa2 = 1.
if (x = 12) rsa3 = 1.
if (x = 13) cg_op = 1.
if (x = 14) cg_cio1 = 1.
if (x = 15) cg_afp23 = 1.
if (x = 16) cg_ft = 1.
if (x = 17) cg_oth = 1.
if (x = 18) cg_dk = 1.
if (x = 19) cg_ia = 1.
if (x = 20) jibnjcct = 1.
if (x = 21) onchnd = 1.
if (x = 22) hnchnd = 1.
if (x = 23) bt_ngcd = 1.
if (x = 24) bt_hhncd = 1.
if (x = 25) othtbq = 1.
if (x = 26) fprofq = 1.
if (x = 27) pprofq = 1.
if (x = 28) nurse = 1.
if (x = 29) cert = 1.
if (x = 30) hdippgce = 1.
if (x = 31) degree = 1.
if (x = 32) pgraddip = 1.
if (x = 33) hdegree = 1.
end repeat.
end if.

```

\* frequencies variables=noquals cse2\_5 cse1 oleva\_c gcsea\_c alev scotoa\_c scots1\_3 scothg scot6th  
rsa1 rsa2 rsa3 cg\_op cg\_cio1 cg\_afp23

\* cg\_ft cg\_oth cg\_dk cg\_ia jibnjcct onchnd hncchnd bt\_ngcd bt\_hhncd othtbq fprofq pprofq nurse cert  
hdippgce degree pgraddip hdegree .

\*\*\*\*\*  
\*\*\*\*\*

\*\*making highest qualification variables to best match 2000 data categories.  
\*\*a levels are treated as equivalent to 2 plus a levels in 2000 as the number of a levels not collected in  
NCDS5.

\*\*\*\*\*  
\*\*\*\*\*

\*\*\*\*\*

\*\*highest academic qualification to match 8 category HIACA00.  
\*\*a levels in ncds5 are treated as equivalent to 2 plus a levels in 2000.

\*\*\*\*\*

compute hacad33b = 0.  
if (any(1,cse2\_5)) hacad33b = 2.  
if (any(1,cse1,oleva\_c,gcsea\_c,scotoa\_c,scots1\_3)) hacad33b = 3.  
if (any(1,alev,scothg,scot6th)) hacad33b = 5.  
if (any(1,cert,hdippgce)) hacad33b = 6.  
if (any(1,degree,pgraddip)) hacad33b = 7.  
if (hdegree = 1) hacad33b = 8.  
if (missing(noquals)) hacad33b = -1.  
missing values hacad33b (-1).

variable labels hacad33b 'ncds33: highest academic qualification to match HIACA00'.  
value labels hacad33b 0'no quals' 2'cse2-5' 3'good o levels' 5'a levels' 6'diploma' 7'degree PGCE'  
8'higher degree'.  
format hacad33b (f3.0).  
\* frequencies variables=hacad33b.

\*\*\*\*\*

\*\*highest academic qualification variable - NVQ levels - to match NVQACA00.

\*\*\*\*\*

recode hacad33b (0=0) (2=1) (3=2) (5=3) (6,7=4) (8=5) into hacnvq33.  
variable labels hacnvq33 'ncds33: highest NVQ level qual from academic qualification to match  
NVQACA00'.  
format hacnvq33 (f3.0).  
\* frequencies variables=hacnvq33.

\*\*\*\*\*

\*\*highest academic qualification variable - ghm measure to match GHMACA00.

\*\*\*\*\*

recode hacad33b (0=0) (2=1) (3=2) (5=3) (6=4) (7=5) (8=6) into hacghm33.  
variable labels hacghm33 'ncds33: highest academic qualification - ghm measure to match GHMACA00'.  
value labels hacghm33 0'no quals' 1'cse2-5' 2'o levels' 3'a levels' 4'sub degree' 5'degree' 6'higher  
degree'.  
format hacghm33 (f3.0).  
\* frequencies variables=hacghm33.

```
*****
**highest vocational qualification at 33 - including nursing - to NVQ levels - to match HIVOC00'.
*****
```

```
compute hvoc33 = 0.
if (any(1,rsa1,othtbq,cg_oth,cg_dk)) hvoc33 = 1.
if (any(1,rsa2,cg_op,cg_cio1,cg_ia,jibnjcct)) hvoc33 = 2.
if (any(1,rsa3,cg_afp23,onchnd,bt_ngcd)) hvoc33 = 3.
if (any(1,cg_ft,hnchnd,bt_hhncd,fprofq,pprofq,nurse)) hvoc33 = 4.
if (missing(noquals)) hvoc33 = -1.
missing values hvoc33 (-1).
format hvoc33 (f3.0).
variable labels hvoc33 'ncds33: highest vocational qualification - to match HIVOC00'.
```

```
* frequencies variables=hvoc33.
```

```
*****
**highest academic or vocational qualification - to NVQ levels - to match HINVQ00.
*****
```

```
compute hq33a = 0.
if (any(1,cse2_5,rsa1,othtbq,cg_oth,cg_dk)) hq33a = 1.
if (any(1,cse1,oleva_c,scotoa_c,scots1_3,rsa2,cg_op,cg_cio1,cg_ia,jibnjcct)) hq33a = 2.
if (any(1,alev,scothg,scot6th,rsa3,cg_afp23,onchnd,bt_ngcd)) hq33a = 3.
if (any(1,cg_ft,hnchnd,bt_hhncd,fprofq,pprofq,nurse,cert,hdippgce,degree,pgraddip)) hq33a = 4.
if (hdegree = 1) hq33a = 5.
if (missing(noquals)) hq33a = -1.
missing values hq33a (-1).
format hq33a (f3.0).
variable labels hq33a 'ncds33: highest NVQ qualification - academic or vocational - to match HINVQ00'.
```

```
* frequencies variables=hq33a.
```

```
*****
**COMBINING NCDS33 AND 2000 DATA VARIABLES.
*****
```

```
*****
**combined highest academic qualification variables - 8 category.
*****
```

```
compute hqaca00 = HIACA00.
if (HIACA00 = 0 and hacad33b >=0) hqaca00 = hacad33b.
do if (HIACA00 > 0 and hacad33b >= 0).
if (hacad33b < HIACA00) hqaca00 = HIACA00.
if (hacad33b = HIACA00) hqaca00 = HIACA00.
if (hacad33b > HIACA00) hqaca00 = hacad33b.
end if.
```

```
variable labels hqaca00 'Highest Academic level up to 2000'.
```

value labels hqaca00 0'no quals' 1'bad o levels' 2'cse2-5' 3'good o levels' 4'as levels or 1 a level' 5'a levels' 6'diploma' 7'degree PGCE' 8'higher degree'.

format hqaca00 (f3.0).

\* frequencies variables=hacad33b hiaca00 hqaca00.

```
*****
**combined highest academic qualification variables - NVQ levels.
*****
```

```
compute hqanvq00 = nvqaca00.
if (nvqaca00 = 0 and hacnvq33 >= 0) hqanvq00 = hacnvq33.
do if (nvqaca00 > 0 and hacnvq33 >= 0).
if (nvqaca00 > hacnvq33) hqanvq00 = nvqaca00.
if (nvqaca00 = hacnvq33) hqanvq00 = nvqaca00.
if (nvqaca00 < hacnvq33) hqanvq00 = hacnvq33.
end if.
```

variable labels hqanvq00 'Highest NVQ level (academic) up to 2000'.

format hqanvq00 (f3.0).

\* frequencies variables=hqanvq00 nvqaca00 hacnvq33 .

```
*****
**combined highest academic qualification variables - ghm measure .
*****
```

```
compute hqaghm00 = ghmaca00.
if (ghmaca00 = 0 and hacghm33 >= 0) hqaghm00 = hacghm33.
do if (ghmaca00 > 0 and hacnvq33 >= 0).
if (ghmaca00 > hacghm33) hqaghm00 = ghmaca00.
if (ghmaca00 = hacghm33) hqaghm00 = ghmaca00.
if (ghmaca00 < hacghm33) hqaghm00 = hacghm33.
end if.
```

variable labels hqaghm00 'highest academic qualification - ghm measure - combining hacghm33 and ghmaca00'.

value labels hqaghm00 0'no quals' 1'cse2-5' 2'o levels' 3'a levels' 4'sub degree' 5'degree' 6'higher degree'.

format hqaghm00 (f3.0).

\* frequencies variables=hacghm33 ghmaca00 hqaghm00.

```
*****
**highest vocational qualification.
*****
```

```
compute hqvoc00 = hivoc00.
if (hivoc00 = 0 and hvoc33 >= 0) hqvoc00 = hvoc33.
do if (hivoc00 > 0 and hvoc33 >= 0).
if (hivoc00 > hvoc33) hqvoc00 = hivoc00.
if (hivoc00 = hvoc33) hqvoc00 = hivoc00.
if (hivoc00 < hvoc33) hqvoc00 = hvoc33.
end if.
```

```
variable labels hqvoc00 'Highest NVQ level (vocational) up to 2000'.
format hqvoc00 (f3.0).
* frequencies variables=hvoc33 hivoc00 hqvoc00.
```

```
*****
**highest academic or vocational qualification - to NVQ levels.
*****
```

```
compute hqnvq00 = hinvq00.
if (hinvq00 = 0 and hq33a >= 0) hinvq00 = hq33a.
do if (hinvq00 > 0 and hq33a >= 0).
if (hinvq00 > hq33a) hqnvq00 = hinvq00.
if (hinvq00 = hq33a) hqnvq00 = hinvq00.
if (hinvq00 < hq33a) hqnvq00 = hq33a.
end if.
```

```
variable labels hqnvq00 'Highest NVQ level (academic or vocational) up to 2000'.
format hqnvq00(f3.0).
frequencies
  variables=hq33a hinvq00 hqnvq00.
```

```
frequencies
  variables=hqaca00 hqanvq00.
```

\* finally, save file of relevant derived education variables from this sweep and NCDS5, to help with future derivations (i.e incorporating NCDS7 and NCDS8).

```
save outfile='D:\Highest_qualis_NCDS5_NCDS6.sav'/keep=ncdsid hacad33b hacnvq33 hvoc33 hq33a
hiaca00 nvqaca00 hivoc00 hinvq00 hqaca00 hqanvq00 hqvoc00 hqnvq00.
```

## **Centre for Longitudinal Studies**

Institute of Education

20 Bedford Way

London WC1H 0AL

Tel: 020 7612 6860

Fax: 020 7612 6880

Email [cls@ioe.ac.uk](mailto:cls@ioe.ac.uk)

Web <http://www.cls.ioe.ac.uk>