

TO EDIT OR NOT TO EDIT: THAT IS THE QUESTION

Some Thoughts on the Current State of Computer Editing

Michael J. Levin
Population Division
U.S. Bureau of the Census
Washington, D.C. 20233

DRAFT

Paper presented for Thirteenth Population Census Conference, December 10 to 14, 1990, East-West Population Institute, Honolulu, Hawaii. Do not cite or quote.

We find the arguments in favor of editing unconvincing. First, "not stated" values and anomalies in census tabulations are valuable indicators of data quality. They facilitate rather than impair use of the data. The argument that editing serves user convenience is fallacious. We are sometimes inconvenienced by not stated values and anomalies, to be sure. The fallacy lies in supposing that the inconvenience is eliminated by editing. The laws of a nation may inconvenience its citizens, but they are not revoked on this account. It is recognized that, despite inconveniences, they serve a necessary function. The same applies to not stated values and anomalies in census tabulations.

Second, editing destroys information. When a not stated value is imputed, the information that no response was obtained for this person is destroyed. When an anomaly is suppressed by changing a value, the original value is destroyed. The argument that editing and imputation improve data quality is fallacious. The effects of editing are cosmetic rather than corrective. Editing conceals defects without eliminating them (Banister and Feeney 1979).

Census and survey editing entralls statisticians and demographers. Many of these statisticians and demographers, as those cited above, condemn the use of any editing in census and survey work. These individuals feel that editing does violence to a data set. The questions here involve whether census editing should be done, and, if so, how much editing is the correct amount.

Both subject matter experts and programmers bring specialized skills to the complex issues revolving around census edits. The arrival of high speed, very "user-friendly" computers has only enhanced and magnified this fascination. If any editing is appropriate, we must then decide how much editing is appropriate. As an ESCAP official noted recently, "Lack of basic editing policy and clear lines of responsibility has sometimes led to confusion and as a consequence the timeliness of reports and quality of data have suffered."

Census and survey editing does not improve the quality of collected data. Edited census data are at most only as accurate as the collected data, and frequently, much the worse for wear after the editing process. If a 70 year old female is recorded as the mother of a 3 year old child, we know something is wrong -- either the mother's age is wrong, or the child's age is wrong (or both). In changing one age or the other, without any other information, about half the time we will have changed correct information to being incorrect. The other half of the time we will have 'improved' the quality of the data. Many times we will not know whether we have attacked the right variable, and

whether we have made the relationship between items better or worse. Demographers and statisticians can debate whether we should change either age. A table of mother's age by child's age would look strange with the 70 year old mother and 3 year old child. Much of census and survey editing, then, is as much an exercise in aesthetics as attempts to improve quality.

Clearly, we want to change the data set as little as possible. Maintaining the integrity of the data is the highest priority. Since edits always introduce error into the data set while trying to eliminate other errors and inconsistencies, the first tenet of editing philosophy must be to try to reduce the introduced error as much as possible.

The largest problem with the neo-mechanization of editing is the loss of timeliness if firm control is lacking. That is, sometimes systems analysts get carried away. The edit becomes so complex that the package is not only not ready when keying the data, but we also lose time just getting the program prepared for edit analysis. Sometimes the logic becomes so convoluted that neither the subject analyst nor the programmer can decipher whether the appropriate and proper paths are followed. That is, there is simply too much editing.

The new editing packages -- like CONCOR and PC-EDIT -- have helped in this revolution in increasing sophistication of computer edits:

The principal advantage in using such packages is that the need to expend scarce resources on systems design is largely eliminated. The user-friendliness of the packages has enabled subject-matter statisticians to control the entire editing operation. This has resulted in appreciably speedier processing and has encouraged greater flexibility in the design of edit systems. While these developments have been most beneficial on balance, they have brought with them new kinds of problems.

By simplifying the tasks of writing edits and by enhancing the role of the statistician, the introduction of the software packages has sometimes resulted in overly complex and lengthy edit specifications as newly acquired skills are demonstrated. In addition the concentration of the edit system in the hands of a single statistician (though sometimes more) familiar with the software, can have disastrous consequences where control is lacking. (ESCAP n.d.:11)

We must build controls into the editing part of the census process, as in any other segment of the process. The fully automated system may be a modern electronic marvel, but if we cannot process the data quickly and accurately, the machine may be shiny, but may not work properly. Until

recently, most post-enumeration census processing required returning to the original forms to reconcile errors or inconsistencies. Even if returning to the forms resolved many issues, this procedure can not resolve respondent and enumerator errors.

Also, improved software and machine capabilities have been accompanied by greater confidence in the computer's ability to detect error and the use of automatic correction. Of course, in speaking of "automatic correction" we obviously mean "automatic change" because, usually, we can not tell whether we have 'corrected' anything. In fact, as noted before, we have really only made the data set more aesthetic, not 'better.'

Some problems are problems only because of faulty thinking and not because of the automatic correction, but become magnified when subject matter people do not properly check edit paths before the census. ESCAP notes two such problems in their review:

...Whenever education was not stated on the census schedule, values were obtained from the "hot-deck" of valid records. But since the failure to respond resulted largely from the enumerators' difficulty in recording 'nil' education for persons who never attended school, the described procedure clearly resulted in a strong upward bias. In yet another country, bias of a different kind was introduced as the imputation procedure amended sex for records that had failed an edit, on the basis of whether or not fertility had been reported. Since in many instances respondents were below reproductive age, systematic errors were introduced, especially in the youngest age groups (ESCAP, n.d.:11).

Of course, both these problems are not the fault of automatic correction but of faulty thinking. The authors note, "The obvious solution is to recognize the potential uses of the technique and ensure that controls are tight, especially by setting tolerances for failure of any edit at very low levels."

So, the main questions become "Do we edit census and survey data?" and "If so, how much?" Here, we assume that an aesthetic product is the major goal of a census. Since the results of a census are completely dependent on the quality of the enumeration, census editing, when done, is done to achieve that aesthetic aim. Modern computer editing requires both subject-matter and programmer expertise. Unfortunately, in many census offices, subject matter specialists and programmers still do not 'speak' a common language. Therefore, here we discuss methods of increasing communication between subject matter specialists and programmers during development of computer edit

specifications. Subject matter specialists and programmers have not traditionally used a common computer package or language to 'talk' to each other. This failure to communicate has resulted in misunderstandings at best, and data problems at worse.

Census editing has many phases. Enumerators manually edit data, as do field operations supervisors, and central office personnel. Then, staff code the data, with further editing resulting, depending on choices coders make on their own, or supervisors force them to make. Then keyers key data, often with edits built into the data entry package. Finally, the data are computer edited, most often now, by a computer software package. Here we look at the automated editing process.

KEYING CONSIDERATIONS

Contemporary data entry packages -- ENTRYPOINT 90 and CENTRY, for example -- build flexibility into the system. The flexibility brings at least two major changes to census keying - (1) skip patterns which were already in previous packages can be much more sophisticated, and (2) data can be edited when keyed.

Beware of skip patterns. The most important part of the keying operation is the keying itself. The most efficient keying is "heads down" keying. That is, the keyer can work best when she or he simply keys the data as received. Whenever the keyer has to stop to check an error or inconsistency, the keying operation inevitably slows down, is less efficient, and more prone to errors. Also, the keyers should never make subject matter decisions.

Sometimes the subject matter persons devising the keying instructions cause these problems themselves. Rather than straight keying of exactly what is on the questionnaire, sometimes specialists program edit decisions and skip patterns into the keying. Consider the following married couple household:

Person	Relationship	Sex	Marital Status	Children
1	Householder	M	Married	10
2	Spouse	M	Married	Blank

If sex takes precedence over Children ever born -- likely, since the keyer would key item 3 (sex) before item 20 (fertility) -- then the skip pattern will delete information for children ever born by the skip pattern. That is, for males, the keying program will skip the fertility information since none should be there. Effectively, we would lose those data, and they would be lost forever.

Later, during computer edit, we would find a same-sex married couple. It is very likely that the second person's sex will change to female (based on similar households in the area). Then the edit will impute her fertility. Therefore, we will not only have deleted fertility information from one person (whose sex should change), we also add inappropriate fertility information to someone else.

The answer to this problem is to let the keyers key, and use the computer editing package properly. The keyer will not slow down by having to stop to figure out what is going on with information that is clearly there but not keyed. Also, we will not lose information that we may need later.

Actually, the U.S. Bureau of the Census International Statistics Programs Center (ISPC) has recently made additions to its CENTRY entry package which allow increased flexibility in keying operations. Computer edits in CONCOR can be written to detect errors in questionnaires as soon as keyers key those questionnaires. These errors are not "corrected". The package notes the problem for the keyer, but more importantly, for a subject matter person. That person can then make changes to the questionnaire early on. Usually, two edit programs would be necessary -- a short one to point out the obvious invalid values and inconsistencies, and a longer one for a more complete edit, the length depending on the complexity specified by the subject matter specialists.

CENTRY has features to aid in this analysis. The keyer can key the data -- all of the data -- as recorded on the questionnaire. Another person, a subject matter person, can then run the CENTRY

program through a short CONCOR program, or, even better, use the 'modify' mode of CENTRY to check entered data with the CONCOR program. Staff can correct obvious errors on the spot.

WHITHER WE GO EDITING?

After keying the data, the problem continues -- how much editing is enough? Since I am suggesting that the first priority for publications is aesthetics, some editing is necessary for some items to eliminate invalid and inconsistent values. When you choose to edit data influences the final product as much as how you edit the data. For example, in the following table we did not edit unknown entries for age and sex:

Table XXXX . Population by Age Group and Sex: Year

Age group	Total	Males	Females	Unknown
Total..	100	48	50	2
< 5 years..	10	5	5	0
5 to 9 yrs.	10	5	5	0
		.		
		.		
75 + yrs...	10	5	5	0
Unknown....	2	1	1	0

In this example, 48 males and 50 females made up 98 percent of the population. The other two persons had unknown sex, and were not edited.

Most planners and policy makers would look at this table, and would say, "Well, since we have two unknowns, one must be male and one female, making the population 49 percent male and 51 percent female." That guess would be as good as any. But, since the only information available for assigning the unknowns is the sex distribution, the imputation uses only that skimpy information. That is, the subject matter specialists who wanted to preserve the unknowns in the data set rather than assign them can do that, but the planners assigned values anyway without any other information - without looking at whether the person had children ever born, or was someone's mother, or someone's

husband. It is important to remember, also, that original values as well as imputed values can be kept on the microcomputer records. A question develops: is it better to allocate when the data are available during the edit procedure, or better to allocate at the end, when no other information is available?

GENERAL COMPUTER EDITING

The proverbial better mousetrap is here in the form of a better edit package. Over the last decade, U.S. Bureau of the Census' International Statistics Programs Center (ISPC) developed CONCOR. CONCOR (CONsistency and CORrection) is a generalized computer package for editing census and survey data. Subject matter specialists and programmers can learn the few important CONCOR commands, and then use these commands with appropriate edit logic to communicate.

Microcomputers have revolutionized data processing, generally, and for this presentation, at least, census and survey processing in particular. For tabulations, for example, no longer must we check large quantities of data by hand, or, wait for hours or days to receive tabulations from a mainframe computer. Now we are able, through a small set of instructions, to get tabulations quickly and with as many breaks as we want.

Similarly, during computer edit, we can check large quantities of data and correct errors quickly. Manual census and survey editing used to take time and was subject to human error. Computer editing reduces both time and the chance of introducing human error. Computer edits check the validity of entries by examining them to see if they have acceptable values. We also can check the value of the entry against related entries for consistency.

We cannot always refer to original documents to correct errors for large volumes of data. Often the data recorded on the original questionnaires are wrong or inconsistent. Computer editing systems like CONCOR can correct erroneous data immediately. Error listings and assigned variables record all errors found and all changes made. Organizations plan computer edits carefully because

running large quantities of data through a computer system is time-consuming. The programmer should plan and design a computer edit to inspect the data and have the computer change them.

Inexpensive microcomputers make computer editing much more "user-friendly." Once programmers write and test the program, CONCOR edits about 4,000 census records per minute on an IBM PC/AT or compatible. Until recently, edit programs had to be custom-written, requiring expensive debugging and processing time. Programmers can develop CONCOR edits rapidly.

Neither CONCOR nor any other editing package or program solves all editing problems. Subject matter specialists still must make decisions about actual editing. Imputation probably introduces more errors into the data set than it removes in very small data sets. CONCOR can easily do imputation if needed, either automatically from a table of values, or by using a constantly changing 'hot deck'. Dumping and printing records using the hot deck is less efficient than simply correcting records with errors. Only in very small populations can staff recheck the original census data to be sure that the collected information is accurate. Changes made are not 'corrections' since, if the attributes of the variables are inconsistent, the changed attribute may introduce yet another error or inconsistency.

The primary advantage of an editing package like CONCOR is that when using the package properly, the data will be consistent and 'clean' so tabulations can be more timely. Most editing can be done more quickly with CONCOR than with custom-written programs because CONCOR does not require the same level of programming knowledge.

The purpose of editing is to make the data as nearly represent real life as possible by cutting omissions and invalid entries, and changing inconsistent entries. Below are some major principles:

1. Keep 'not reported' for certain items. Thus, for an omission or an inconsistent, impossible, or unreasonable entry, the edit assigns 'not reported'.
2. Make the fewest required changes to the originally recorded data.
3. Eliminate obvious inconsistencies among the entries.

Supply entries for erroneous and missing items by using other entries. These entries may be for the housing unit, person, or other persons in the household or comparable group as a guide. Always follow specified procedures.

Methods of changing data. Actual methods of correcting (i.e., changing) vary depending upon the item. Usually, data items are assigned valid codes with reasonable assurance that they are correct by using responses for other data items within the record, or in other records in the questionnaire.

When recorded responses are missing, impossible, inconsistent, or unreasonable and cannot be determined from other responses in the same questionnaire, some technique must assign entries.

The edit procedure can give a particular response for each occurrence of unknown entries, or a procedure can impute responses proportionally from a distribution of responses. We call this procedure the 'cold deck' method.

In the cold deck method, the edit does not update the original array. The cold deck values would not change from those in the starter deck after the first, second, tenth or any other person record. The edit assigns the original values for any allocations of missing data.

For proportional distribution of responses, suppose a tabulation of valid data on hours worked per week by males 33 years old employed in agriculture showed that 25 percent worked 50 hours a week, 40 percent worked 60 hours a week, and 35 percent worked 70 hours a week. Missing or invalid responses for hours worked for males 33 years old employed in agriculture would be replaced 25 percent of the time by 50 hours, 40 percent of the time by 60 hours, and 35 percent of the time by 70 hours. Unless reliable data are available from previous censuses, surveys, or other sources, this technique requires pre-tabulation of valid responses from the current census, which may not be economically or operationally feasible. Otherwise, replacement of invalid data may result in the production of inaccurate statistics. Part of the Integrated Microcomputer Processing System (IMPS) - QUICKTAB -- helps in this by giving unedited and edited distributions.

Hot deck technique. Unknown data occur in all censuses and surveys. The missing data may be due to informant error, enumerator error in hearing or recording, or to coding, editing, or punching errors. If the enumerated data are readily available and time and money permit, coding, editing, or punching errors are simple to correct. We cannot correct informant and enumerator errors so easily; staff either have to contact the original respondent to correct the aberrant information, or the information remains 'unknown'.

In compiling and displaying census or survey data, we have a column or row showing the number of unknowns for a particular data item. Carrying this information along can be very cumbersome, especially since there will be different individuals and numbers of individuals for different items. Therefore, the number of persons for whom data is available varies from table to table. Procedures provide the information missing from data and to avoid discrepancies and the need to figure out percentages twice, with and without the unknowns. One method of ridding the data of unknowns is the use of 'hot deck'.

Hot decks allocate a value when it is unavailable, unknown, incorrect or inconsistent (and must change). The hot deck approach uses known information about individuals with similar characteristics (for example, sex, age, relationship). These characteristics help to find the 'most appropriate' information when some piece (or pieces) of related information for other individuals is unknown. The hot deck itself is a set of values, like cards in a deck, used to store, and then to provide, information for unknown values. The deck constantly changes, being systematically shuffled, so responses change as the edit processes the data.

At its simplest, a single variable is the 'deck'. For sex, for example, we assign an initial value (male or female) to the deck arbitrarily, thus determining the 'seed'. Then, if a person's sex is blank for some reason, the seed value becomes the sex of the first individual with unknown sex. If we know the first person's sex, the sex of that person replaces the seed value. If the second person's sex is unknown, the sex stored in the hot deck (the sex of the first person) becomes the sex of this person.

Below is some sample information for a set of ten individuals. The numbers 9 for sex and 99 for age show missing information. Although other variables are available for use in allocation (e.g., education, occupation), this short example does not include them.

Person	Relationship	Sex	Age
1	1	1	39
2	2	2	35
3	3	1	13
4	3	9*	10
5	4	2	40
6	4	1	99 *
7	4	2	13
8	5	9*	99 *
9	5	1	44
10	5	2	36

We can use relationship to householder and sex to aid in determining the age to be assigned to an individual. That is, we use a two-dimensional array instead of a single variable. Assume we have the following list of relationship codes:

- 1 = Householder
- 2 = Spouse
- 3 = Child
- 4 = Other relative
- 5 = Nonrelative

We can create a starter deck with age values that might approach the real situation for the relationship be sex we are considering. The values in the starter deck, as noted elsewhere, are not very important since they are almost certain to be replaced before being called for use. Also, if the hot deck assigns enough values, the few initial values in the starter deck used will not affect the final tabulations very much. The starter deck values might be like these:

SEX	RELATIONSHIPS				
	Householder (1)	Spouse (2)	Child (3)	Other (4)	Nonrelative (5)
M(1)	35	35	12	40	40
F(2)	32	32	12	37	37

Since the first person in our sample is a householder (code = 1) and he is male (code = 1), his age (39) replaces the first element (coordinates 1,1) during the hot deck allocation. The deck then contains the following values:

SEX	Householder (1)	RELATIONSHIPS			
		Spouse (2)	Child (3)	Other (4)	Nonrelative (5)
M(1)	39*	35	12	40	40
F(2)	32	32	12	37	37

The second person is spouse (code = 2) and female (code = 2), so her age (35) replaces the value in the second row of the second column, changing the deck to these values:

SEX	Householder (1)	RELATIONSHIPS			
		Spouse (2)	Child (3)	Other (4)	Nonrelative (5)
M(1)	39*	35	12	40	40
F(2)	32	35*	12	37	37

The ages of other individuals in the household similarly replace starter or subsequent values as we encounter them. After the fifth person, we find the following situation:

SEX	Householder (1)	RELATIONSHIPS			
		Spouse (2)	Child (3)	Other (4)	Nonrelative (5)
M(1)	39*	35	13*	40	40
F(2)	32	35*	12	40*	37

We see that the hot deck changed four of the initial values. Note that the edit assigns person 4 sex 1 by the previous sex allocation procedure. Also, because the edit imputed a value for sex, we do not update the array with that person's age. We will update only with values from records when sex and relationship are both initially correct. When we get to person 6, we find that the age is unknown. We do know that the person is male, and he is an 'other relative' of the householder. We therefore look in the hot deck element for males whose relationship is 'other relative' (that is, the

fourth column in the first row), and assign the value of age for that category ('male other relative' -- here, 40).

Neither sex nor age is known for the eighth person. The edit allocates female. Then, the hot deck allocates age based on this allocated sex and the relationship code (5). Here the age is 37.

Although we have allocated the value for age from the known relationship, we have used a previously allocated value for sex for the other dimension of the matrix. This use of allocated values for further allocation is a poor editing procedure. It would be better to look for other known data items (e.g., marital status) for use in the allocation.

After the tenth person, the hot deck values are these:

SEX	Householder (1)	RELATIONSHIPS			
		Spouse (2)	Child (3)	Other (4)	Nonrelative (5)
M(1)	39*	35	13*	40	44*
F(2)	32	35*	12	13*	36*

In this example we only use one starter value in the allocation. Usually the edit uses few initial values in allocation, but most cases result in values assigned from the population.

How hot decks work. The following figure shows part of the starting values for a hot deck to obtain the number of children ever born based on marital status, household relationship, and age of mother:

47	MEDIUM M20-FERTILITY (72,5)						
48						
49						
50	Marital and Household Status						
51						
52						
53						
54						
55	Now	Widowed/		House-	house-	Not a	Hispanic
56	Married	Divorced	Separated	holder	holder	Origin	and
57						
58						
59						
60							HISPANIC
61	1	1		1	1	1	15
62	1	1		1	1	1	16
63	2	1		2	2	1	17
64	2	1		2	2	1	18
65	3	1		2	2	1	19
66	3	1		2	2	1	20
67	3	1		2	2	1	21
68	3	1		2	2	1	22
69	3	1		2	2	1	23
70	3	2		2	2	1	24
71	3	2		3	2	1	25-29
72	4	3		3	2	1	30-34
73	4	3		3	2	1	35-39
74	5	4		4	2	2	40-44
75	5	4		4	2	2	45-49
76	5	4		4	2	2	50-59
77	5	4		4	2	2	60-69
							70+

Note that the hot deck has titles for both the rows and columns. The titling for the rows (the age groups) appears on the right because CONCOR reads from left to right. CONCOR reads the titles as initial values, if titling were on the left. As it is, the initial values are easy to identify.

Initial values usually come from some source other than the census itself. These initial values used to concern subject matter specialists, since, if the hotdeck values are not 'updated' before use, the edit uses initial values. Subject matter specialists often do not trust these values. About 2 percent of the time, the edit uses these initial or 'cold' deck values. Since no data are initially available, it would be difficult to assign initial values. Sometimes subject matter specialists use data from a previous census for initial values, sometimes data from an intercensal survey, sometimes from administrative records or another source. Census offices now often use microcomputers to edit census data. It is now possible to assign any values initially, then run through a large segment of the data, saving the 'updated' values. These values then become the initial values; we then rerun the edit. This

procedure does not eliminate all errors. The hotdeck could still donate a cell's value before being replaced by a hotdeck value, but the procedure reduces chances of pulling an initial value.

Also, in any hot deck procedure, we must obtain the distribution of allocated and unallocated values to make certain that the logic used is working properly. A subject matter specialist may not always want the distributions to look the same. For example, if all persons under 15 are to be "never married," then we would not expect a distribution of allocated values for marital status for these young persons to be the same as for adults. The distribution of allocated values for marital status for adults (unless the householder or spouse determines the other) should be about the same as the unallocated values.

Similarly, in the example given above, the distribution of allocated numbers of children ever born by age of mother should be about the same as the unallocated values.

The example shows a two dimensional matrix. For the fertility item, these dimensions are probably enough. There are other cases when problems develop from this fairly simple procedure. For example, if a parent does not know the age of one of his or her children, then probably the parent gives ages for none of the children. If a household has four children, none having age, then the edit might call the same cell each time. The family could end with quadruplets! With two children of each sex, and sex is one dimensions in the hotdeck, you get two sets of twins. The United States Bureau of the Census eliminates this problem partially by keeping eight values in each cell, and cycling through them, both for allocating and for updating. A series of pointers keeps everything straight.

Display formats. I developed some CONCOR edits from the Edit Specifications for the 1990 Census of the United States. I found that the display of CONCOR code can aid in interpreting the flow of the edit. CONCOR is a means of communication between the programmers and the subject matter specialists. This communication takes place both at the time of the census edit, and later. The communication helps adapt edits for intercensal surveys, and for use in developing edits for subsequent censuses.

The example below shows part of the fertility edit for the 1990 Census of the United States:

Line	Code	Text	Sex	Age	RESULT
248	SUBROUTINE	S-FERTILITY			
249				
250		Age and sex break			
251				
252				Over	
253	IF	SEX = 2	FEMALE	14	CALL
254	AND	AGE > 14		years	SUB-FERT
255		CALL SUB-FERT		old	
256	ELSE				
257	IF	SEX = 1			
258	ASSERT	p20 = BLANK			
259	MSG	"MALE WITH CHILD"			
260	P00	SEX AGE P20			Blank out
261	FAIL		Male	(X)	fertility
262	ALLOC	P20 .Male with child			
263		= BLANK			
264	MSG	"P20:MALE WITH CHILD"			
265	LIST	"MALE WITH CHILD" P20			
266	WRITE	"MALE WITH CHILD:"			
267		" SEX = " N1 P03			
268		" CEB= BLK"			
269	END-FAIL				if child
270	ELSE				is present
271	ASSERT	P20 = BLANK msg			
272	MSG	"YOUNG FEMALE WITH CHILD"			
273	P00	SEX AGE P20		Under 15	
274	FAIL		Female	yrs old	
275	ALLOC	P20 .Female too young			
276		= BLANK			
277	MSG	"P20:FEMALE TOO YOUNG"			
278	LIST	"FEMALE TOO YOUNG" P20			
279	WRITE	"FEMALE TOO YOUNG:"			
280		" AGE =" N3 P05A			
281		" CEB= BLK"			
282	END-FAIL				
283	END-IF				
284	END-IF				
285	END-SUBR				

Follow the logic on the right side of the example. The columns form something of a decision chart. The last column always tells you what the edit does, depending on the previous columns. The first column looks at sex, and the second looks at age within sex. So, if the person is female AND she is more than 14 years old, we will call a subroutine called SUB-FERT (for SUBroutine for FERTility.) If the person is male (and has fertility), the edit blanks fertility; if the person is a female under 15 years old with fertility, the edit blanks fertility for her.

The example also illustrates some guidelines to aid both programmers and subject matter specialists. One useful rule is that no subroutine should be longer than a page. We use a single page for aesthetics, and because that is about what a single thought, a single piece of the decision chart

usually takes. In restricting the subroutine to a single page, we use extensive comments to help in developing the flow of the edit.

A second guide involves avoiding too much 'nesting.' Nesting occurs when a series of "IFs" within other "IFs" develops. When we restrict subroutines to single pages, nested IFs are less likely. Anyhow, the logic is easier to follow when we avoid this kind of nesting. CONCOR has a RETURN command that also helps lessen some need for nesting.

The actual CONCOR code used here is longer than most programmers would like. I have inserted extra commands to help reconcile the subject matter 'logic' with what the CONCOR program does. Three commands, in particular, help subject matter specialists and programmers communicate during CONCOR development and testing. We drop two of the three items during production runs.

MESSAGE DISPLAYS

CONCOR has three commands to help subject matter persons in developing edit specifications. I illustrate the use of these commands in the 1990 United States Census Fertility item example. The commands are:

1. ALLOCATE with MESSAGE
 2. LIST
 3. WRITE
1. ALLOCATE with MESSAGE

An example of the results of use of this command appears below:

CONCOR Version 3.0 Statistics by Edit Command using Dictionary - POA90 90/03/21 07:35:06 Page 1
Area: 0001

Line Number	Message	Number of Allocations	Cases Tested	Cases Failed	Percent Failed
260	MALE WITH CHILD		26	2	7.6
264	P20:MALE WITH CHILD				
273	YOUNG FEMALE WITH CHILD	2			
277	P20:FEMALE TOO YOUNG		13	4	30.77
294	FEMALE WITHOUT FERTILITY	4	39	2	5.13

The ALLOCATE command automatically generates error messages. We add NOSTATS to suppress messages. CONCOR reports messages and cases in order in the REPORT by COMMAND. If we do not include a MESSAGE, the program prints whatever is on the line with the ALLOCATE statement. Frequently, what the edit prints is of little use in analyzing the success of the edit. Therefore, it is a good idea to add a message giving some information about the allocation. Here, we print the first message "MALE WITH CHILD" when a male has recorded fertility - males should be blank in the fertility item.

The REPORT by COMMAND also gives the line numbers in the CONCOR program to help in finding and correcting logic errors. Line 260 contains the MESSAGE with the ALLOCATE statement blanking fertility information for males with children.

We find that the test deck had 2 failed cases of "MALE WITH CHILD"; that is, there were 26 males in the sample, 2 with children, a failure rate of 8 percent. This report is useful in analyzing the edits, both in preparation and in production.

2. LIST

We use the LIST command to get errors by questionnaire. LIST command displays appear in the REPORT by Questionnaire. An example appears below:

CONCOR Version 3.0 Statistics by Questionnaire using Dictionary - POA90 90/03/21 07:35:06 Page 1
Area: 0001

Line Number	Message	Quest: 0001	User Name and Current Value
273	YOUNG FEMALE WITH CHILD		P00 (5) = // SEX = 2 AGE = 10 P20 (5) = 01 P20 (5) = //
280	FEMALE TOO YOUNG		

Line numbers help in correcting logic in the program. A series of displayed items (vertically along the right hand side of the page) helps decide about changes to the program. In this example person 5 is a female, aged 10 years old with 1 child. Edit specifications require blanking fertility for females less than 15 years old. Refer to line number 273 in the program to be sure that the edit is working.

CONCOR uses the LIST command with the IF-THEN-ELSE structure. We also can use the ASSERT statement, but ASSERT only triggers errors for negative results. That is, the statement with the ASSERT must fail to trigger the message. Most subject matter specialists do not write edits negatively, so sometimes a fair amount of work is necessary to reconfigure the edit for CONCOR.

Messages connected with the LIST can be lengthy, but the proposed format here - CONCOR on the left, narrative on the right, prohibits very long messages.

For production runs, we suppress LIST statements, either through commenting out, or removal.

3. WRITE

CONCOR developers did not develop the WRITE command for subject matter use. Since neither of the other commands gives horizontal item listings, we use the WRITE command for this purpose. An example appears below:

FERTILITY PERSON		QUESTIONNAIRE = 0001				RELAT	RACE	HISP	ROW	COLUMN
I:	O:	CEB	SEX	AGE						
0001	0001	00	1	050	00	01	1	0034	0001	
0002	0002	04	2	040	00	01	1	0032	0001	
0003	0003	01	2	017	00	01	1	0021	0005	
0004	0004	00	2	015	00	01	1	0019	0005	
0005	0005	01	2	010	00	01	1	0019	0005	
FEMALE TOO YOUNG: AGE = 010 CEB = BLK										
0006	0006	00	1	006	00	01	1	0019	0005	

The WRITE command gives the item tested (Fertility) and questionnaire number for each questionnaire during testing and prints a line of titles to help in bookkeeping.

In the example, persons 1 to 4 have no fertility inconsistencies. Person 5, has an inconsistency between age and number of children ever born. The edit prints the message "FEMALE TOO YOUNG: AGE = 010 CEB = BLK", showing the error message, and the variables affected. The "I" stands for input and the "O" for output. The second column gives the person number. For this person, the third column "CEB," for children ever born, is inconsistent with the fifth column - age. Females 10 years old cannot have children according to the edit specifications. The edit blanks the number of children ever born.

Note that person 2 has an increase in children ever born between input and output. In the United States, the fertility specialist reserves the code 00 for males and females underage. Therefore, females 15 years and over have 01 if they have no children, 02 for 1 child, etc. The edit makes the change.

Here, the WRITE command gives the column and row of the FERTILITY matrix for helping see whether the hotdeck is working properly. Finally, after testing, we delete WRITE commands.

TEST DECK DEVELOPMENT

It is useful to develop the deck systematically to test the edit logic. For the U.S. edits, I found the most useful way of developing the test deck is the following:

1. First, CENTRY describes a complete household for testing. We run CONCOR (either interactively or separately) to make certain that no errors occur in the initial household. The first household or the program changed until the test household and the program are compatible.
2. Then, we introduce a single error in the test deck to produce the Statistics by command. We replace the error by good data.
3. Next, we reproduce the good household (using CENTRY, or in my case, the NORTON editor) to get some good households.
4. Finally, we introduce errors, case by case, by going down the list of messages in the statistics by edit command.
5. By not numbering the housing units consecutively, it is possible to add households to detect new errors as they develop in the edit process.
6. After testing the edit with the test deck, it is a good idea to comment out LIST and WRITE commands.

CONCLUSIONS

This paper presents some thoughts about the current state of census and survey editing. Microcomputers have revolutionized many aspects of census data processing, census editing more than others. The tools are now available to do very sophisticated editing easily and efficiently. But the computer is just a tool. Subject matter specialists still have the responsibility of deciding what editing to do and how to do it. Subject matter personnel and programmers working together must decide whether to edit census data, and if they are to edit, how much editing is appropriate, efficient, and cost effective. In the end, the most useful, most aesthetic census product should result.

REFERENCES

Banister, Judy and Griffith Feeney
1979 "Census Editing: Is It Overused". Comment made at the Seventh
Population Census Conference, EWPI, Sept 24-29, 1979.

ESCAP
n.d. Editing Policy for the 1990 Population Censuses. Manuscript.