Subject: Re: Problem with dates in the Ethiopia datasets
Posted by Mark on Wed, 01 May 2019 05:56:13 GMT
View Forum Message <> Reply to Message

Dear Sir/Madam,
I am using the 2016 EDHS IR dataset to calculate contraceptive method used prior to the most recent birth.
I am actually interested to calculate it for women who are eligible for birth interval variable (I dropped non-eligible respondents (keep if b11_01!=.)).
Even though I used the command (presented bellow) that should be used for calendar data using Stata software (after reading the 'DHS Contraceptive Calendar Tutorial), I got the highest number of missing data (30.31%) which is not common in DHS data. Is there anything I may miss in my program? May you check my program code, and provide me the right program code for this particular case the 2016 Ethiopian DHS please.

```
* Step 1.1
* length of full calendar string including leading blanks (80)
* actual length used according to v019 will be less
egen vcal_len = max(strlen(vcal_1))
* most calendars are 80 in length, but those without method use may be short, so use the max
label variable vcal_len "Length of calendar"
* Step 1.2
* position of last birth or terminated pregnancy in calendar
gen lb = strpos(vcal_1,"B")
gen lp = strpos(vcal_1,"T")
* update lp with position of last birth if there was no terminated pregnancy,
* or if the last birth was more recent than last terminated pregnancy
replace lp = lb if lp == 0 | (lb > 0 & lb < lp)
* e.g. if calendar is as below ("_" used to replace blank for display here):
*  ____00000BPPPPPPPP00000055555500000TPP00000000000000BPPPPPP PP00000000
*              ^
* lp would be 20
label variable lp "Position of last birth or terminated pregnancy in calendar"
label def lp 0 "No birth or terminated pregnancy in calendar"
label value lp lp
* get the type of birth or terminated pregnancy
* lp_type will be set to 1 if lp refers to a birth,
* and 2 if lp refers to a terminated pregnancy using the position in "BT" for the resulting code
gen lp_type = strpos("BT",substr(vcal_1,lp,1)) if lp > 0
label variable lp_type "Birth or terminated pregnancy in calendar"
label def lp_type 1 "Birth" 2 "Terminated pregnancy"
label value lp_type lp_type
list vcal_1 lp lp_type in 1/5
tab lp lp_type, m
* Step 1.3
* if there is a birth or terminated pregnancy in the calendar then calculate CMC
* of date of last birth or pregnancy by adding length of calendar to start CMC
* less the position of the birth or pregnancy
* calendar starts in CMC given in v017
```

```
* lp > 0 means there was a birth or terminated pregnancy in the calendar
gen cmc_lp = v017 + vcal_len - lp if lp > 0
label variable cmc_lp "Century month code of last pregnancy"
* e.g. if calendar is as below and cmc of beginning of calendar (V017) = 1321:
*  ____00000BPPPPPPPP00000055555500000TPP00000000000000BPPPPPP PP00000000
* cmc_lp would be 1381, calculation as follows:
* 1321 + 80 - 20 (80 is the vcal_len, and 20 is the position of lp)
list v017 lp vcal_len cmc_lp in 1/5
* check the variables created.
tab lp
tab cmc_lp
* list cases where cmc_lp and b3_01 don't agree if the last pregnancy was a birth
list cmc_lp b3_01 if lp > 0 & lp == lb & cmc_lp != b3_01
* there shouldn't be any cases listed.
* Step 1.4
* get the duration of pregnancy and the position of the month prior to the pregnancy
* start from the position after the birth in the calendar string by creating a substring
* indexnot searches the substring for the first position that is not a "P" (pregnancy)
* piece is the piece of the calendar before the birth ("B") or termination ("T") code
gen piece = substr(vcal_1, lp+1, vcal_len-lp)
* find the length of the pregnancy
gen dur_preg = indexnot(piece, "P") if lp > 0
* dur_preg will be 0 if pregnant at the start of the calendar
label variable dur_preg "Duration of pregnancy"
* e.g. if calendar is as below:
*  ____00000BPPPPPPPP00000055555500000TPP00000000000000BPPPPPP PP00000000
*                |12345678^
* dur_preg would be 9 for the last pregnancy (1 B plus 8 Ps)
* if we find something other than a "P" then that is the month before the pregnancy
* if it returns 0 then the pregnancy is underway in the first month of the calendar
* now get the position in the calendar to reflect the full calendar
* not just the piece before the birth, by adding lp
* _bp means 'before pregnancy'. pos_bp means position before pregnancy
gen pos_bp = dur_preg + lp if dur_preg > 0
label variable pos_bp "Position before pregnancy"
label def pos_bp 0 "Pregnant in first month of calendar"
label val pos_bp pos_bp
* e.g. if calendar is as below:
*  ____00000BPPPPPPPP00000055555500000TPP00000000000000BPPPPPP PP00000000
*                 ^
* pos_bp would be 29
list vcal_1 lp dur_preg pos_bp in 1/5
tab dur_preg lp_type, m
* Step 1.5
* find the last code that is not 0 before the pregnancy (using indexnot),
* searching in a substring of the calendar from the month before pregnancy and earlier,
* but not more than 5 years back
* lnz means 'last non-zero before the pregnancy'
```

```
gen lnz = indexnot(substr(vcal_1, pos_bp, vcal_len - pos_bp + 1),"0") ///
  if inrange(pos_bp, 1, vcal_len)
  * get the actual position in the calendar of the last non-zero before the last birth
gen pos_lnz = pos_bp + lnz - 1 if inrange(lnz, 1, vcal_len)
* if last non-zero is more than 5 years before interview, set position to 0
replace pos_lnz = 0 if lnz == 0 | (pos_lnz != . & pos_lnz > v018+59)
label variable pos_lnz "Position in calendar of last non-zero before pregnancy"
label def pos_lnz 0 "No non-zero preceding the pregnancy in the last 5 years"
label val pos_lnz pos_lnz
* list a few cases to check
list vcal_1 lp pos_bp pos_lnz in 1/5
* Step 1.6
* check if the respondent is using a method before the pregnancy but in the last 5 years
gen code_lnz = substr(vcal_1, pos_lnz, 1) if inrange(pos_lnz, v018, v018+92)
replace code_lnz = "0" if pos_lnz == 0
* if the code is NOT(!) a zero ("0"), a "B", "P" or "T" then the respondent was using a method
gen used_bp = !inlist(code_lnz, "0","B","P","T") if code_lnz != ""
label variable code_lnz "Last non-zero code before pregnancy"
label variable used_bp "Using a method before the last pregnancy"
label def used_bp 0 "No" 1 "Yes"
label val used_bp used_bp
* list a few cases to check
list vcal_1 lp pos_bp pos_lnz code_lnz used_bp in 1/5
* Step 1.7
* last method used before pregnancy, but may have been followed by a period of non-use
* converting the string variable to numeric, although it isn't really necessary for most analyses
* set up a list of codes used in the calendar, with each position matching the coding in V312
* use a tilde (~) to mark gaps in the coding that are not used for this survey
* e.g. Emergency contraception and Standard days method do not exist in this calendar
* note that some of the codes are survey specific so this list may need adjusting
scalar methodlist = "123456789WNALCF~M~"
gen method_bp = strpos(methodlist,code_lnz) if code_lnz != ""
* convert the missing code to 99
replace method_bp = 99 if code_lnz == "?"
* now check if there are any method codes that were not converted, and change these to -1
replace method_bp = -1 if method_bp == 0 & used_bp == 1
* alternatively,
* use the do file below to set up survey specific coding using scalar methodlist and label method
* and recode the method and/or reasons for discontinuation
* include the path to the do file if needed
*run "Calendar recoding.do" code_lnz method_bp
* and skip the value labeling in step 2.8 as the do file above includes the value labeling
* if no method was used, set method_bp to 0
replace method_bp = 0 if used_bp == 0
* Step 1.8
* label the method variable and codes
label variable method_bp "Method used before the last pregnancy (numeric)"
label def method ///
```

```
  0 "No method used" ///
  1 "Pill" ///
  2 "IUD" ///
  3 "Injectable" ///
  4 "Diaphragm" ///
  5 "Condom" ///
  6 "Female sterilization" ///
  7 "Male sterilization" ///
  8 "Periodic abstinence/Rhythm" ///
  9 "Withdrawal" ///
 10 "Other traditional method" ///
 11 "Norplant" ///
 12 "Abstinence" ///
 13 "Lactational amenorrhea method" ///
 14 "Female condom" ///
 15 "Foam and Jelly" ///
 16 "Emergency contraception" ///
 17 "Other modern method" ///
 18 "Standard days method" ///
 99 "Missing" ///
 -1 "***Unknown code not recoded***"
label val method_bp method
tab method_bp
```