

Adjusting Dates in a New Century

Malachy J. Foley

University of North Carolina at Chapel Hill, NC

ABSTRACT

The Y2K scare is over, but the problem of manipulating dates correctly is still current. This paper presents SAS code that automatically adjusts the YEARCUTOFF option based on the current year. This "sliding" cut-off date means you will never have to worry about the YEARCUTOFF option again. It also insures that you can always store and manipulate the maximum range of dates.

INTRODUCTION

The YEARCUTOFF= option makes SAS mostly year 2000 (Y2K) compliant. Usually this option is set to a constant year. The constant year can be either the default value that SAS provides or a value the SAS user supplies.

Unfortunately, all constant years have some sort of drawback. The principal drawback is that as the years go by, any constant needs to be updated for the program to continue functioning properly. Furthermore, the version 6 default for the cut-off option is too low for today's (year 2002 and beyond) computing. On the other hand, the version 8 default is too high to accommodate the processing of senior citizens birthdays.

This paper examines the "just right" goldilocks sliding year for the YEARCUTOFF= option. It also examines the manipulation of years in SAS, and how to avoid forever changing the year you use in the cut-off option.

THE Y2K PROBLEM

The first commercial computers had very limited storage capabilities. For example, a large mainframe computer in the 1960's might have 16K bytes of memory. In the 1980's, the first microcomputers typically came with 16K bytes of memory. At the same time offline storage, like diskettes, also had very limited space.

As a result of these space limitations, programmers would represent years by the last two digits of the year. For instance, the year 1982 would be just 82 (the 19 was understood). This technique worked well and saved space.

As the end of the 1900's neared, the two-digit year presented many problems for legacy computer programs. It was apparent that many programs would recognize "00" as 1900 and had no way of representing the year 2000. This situation was called the Year 2000 problem or the Y2K problem.

THE Y2K SOLUTION

In today's world of megabytes and gigabytes, there is almost always sufficient storage space both online and offline to store four-digit dates. Thus when possible, the easiest solution to the Y2K problem is to always use four-digit years (i.e. 1998, 2002, etc.). All forms and questionnaires, whether paper or electronic, should request four-digit years. Similarly, any character variable in a SAS data set should contain the complete four-digit year (rather than a two digit year). Likewise, computer programs should output four-digit years.

For outputting SAS dates, programs often employ the DATE9 or the MMDDYY10 format. Note that these formats default to DATE7 and MMDDYY8 which print two-digit years. While at times it is tempting to use a two-digit year to save space on a printout, it is unwise. The author has seen cases where the first two implied digits (19 or 20) were misunderstood. In one such case, a programmer spent hours trying to figure out why his program was not operating correctly.

Incidentally, the DATE9 format avoids possible confusion between the day and the month that can occur in international settings. DATE9 also utilizes one less character than the MMDDYY10. For these reasons the author recommends the use of the DATE9 format.

While utilizing the four-digit year is the easiest way to solve the Y2K and Y3K problems, sometimes it is not possible to implement. On occasion the programmer just receives a data set with two-digit years. In these cases, you have to be aware of the YEARCUTOFF= option.

THE YEARCUTOFF= OPTION

The YEARCUTOFF= option is a SAS system option. One way to invoke this option is within the OPTION statement. For example,

```
OPTIONS YEARCUTOFF=1910;
```

(The value after the equal sign is a four-digit year.) The YEARCUTOFF= option allows a two-digit year to span two different hundred-year prefixes. For instance, if you specify YEARCUTOFF=1920, then two-digit years of 20-99 are assumed to be prefixed with 19, and 00-19 are prefixed with 20. Another way to look at this option is that it specifies the first year in a one hundred year period. Thus, if 1920 is the first year in the hundred years, then logically 2019 is the last year in the period.

The YEARCUTOFF= option works in conjunction with the various DATE and DATETIME informats and functions. For example, if you have specified YEARCUTOFF=1920 and you have the following statements:

```
DATA _NULL_;  
  CHAR="05/06/87";  
  MYDATE=INPUT(CHAR,MMDDYY8.);  
  PUT "**** " MYDATE DATE9.;  
RUN;
```

Then "**** 06MAY1987" would be printed to the LOG.

Similarly, if you change the second statement to read CAHR="05/06/01" then "**** 06MAY2001" would be printed to the LOG.

DOWNSIDES OF A FIXED YEARCUTOFF=

When YEARCUTOFF= is set to a constant there is always a downside.

For instances, in SAS version 6.12, the default is YEARCUTOFF= 1900. Thus, any two-digit year would be prefixed with 19 in that version. This default is not Y2K compliant. In other words, SAS programs may not function correctly on or after the year 2000.

In SAS version 8, the default is YEARCUTOFF= 1920. The 1920 default is illustrated in the previous section. If you are processing birth dates, the 1920 default requires that a birth year be between 1920 and the current year (say 2002). Someone born in 1919 cannot be processed.

Of course, the solution is to go to four-digit years and avoid the limitations of two-digit years.

If you cannot use four-digit years, you might consider adjusting the YEARCUTOFF= to some year less than 1920 and more than 100 years before today's year, say 1910. Setting the YEARCUTOFF= option to a fixed date like 1910 also has its difficulties. The main difficulty with specifying a fixed year is that at some point that year becomes obsolete and so does your program. As a case in point, if you set YEARCUTOFF=1910, in 2010 your program will be obsolete.

A SLIDING YEARCUTOFF= DATE

One solution to the limitations of a fixed YEARCUTOFF= date is to utilize a sliding date. Namely, rather than setting the year to a constant, the YEARCUTOFF= is set to the results of a calculation based on today's year.

The sliding year can be anywhere up to 99 years before the current year. The author uses 95 years before current year. So if today's year is 2002, YEARCUTOFF= is set to 1907 (2002-95). The code that sets the sliding YEARCUTOFF= option 95 years before the present year is.

Exhibit 1: Setting YEARCUTOFF= to a Sliding Date

```
OPTIONS YEARCUTOFF=  
  %EVAL(%sysfunc(year(%sysfunc(date())))-95) ;
```

(The %sysfunc allows you to utilize DATA Step functions without using the DATA Step).

With YEARCUTOFF= set this way, the program can be run in any year with any version of SAS and always allow you to input dates that are as far back as 95 years from the current year. So if you are processing dates in the year 2002, you can input birthdays as far back as 1907.

Every year, the code given in Exhibit 1 automatically updates itself. This code has two advantages over a fixed year. First, it never has to be changed, even if you are running your code 10 or 20 years from now. Second, it allows for the storage and processing of dates up to 95 years before the process.

CONCLUSION

In the twenty-first century and the gigabyte-computing world, all dates should always include four-digit years. Nonetheless, if you are confronted with a two-digit year, you must be concerned with the YEARCUTOFF= option. Even if you are not confronted with a two-digit year, it is good practice to include the YEARCUTOFF= option in your SAS programs so they will always be ready to properly input or process a stray two-digit year.

The YEARCUTOFF= option defaults to a constant year. However, a constant cut-off year has two problems. First, it will eventually become obsolete, as is the default cut-off of 1900 in version 6.12 SAS. Second, a high constant year can limit the range of dates that can be processed, as is the case with the 1920 default in version 8.2 SAS.

A sliding cut-off year avoids both of these problems. This paper shows a method for calculating a sliding cut-off YEAR. The method is detailed in Exhibit 1 of the previous section.

AUTHOR CONTACT

Malachy J. Foley
2502 Foxwood Dr.
Chapel Hill, NC 27514
Email: FOLEY@unc.edu