

Paper 069-29

A SAS Macro to Count Consecutive Days

Pingping Zhang, Kendle International Inc., Cincinnati, OH

ABSTRACT

Patient diary data is becoming increasingly important in many indications. This paper describes the macro that counts the number of consecutive days among a set of dates. If a number is supplied, say *n*, then in the resultant data set, we will see subsets of consecutive days with their respective first and last dates listed for each subject if the number of consecutive days is *n* or bigger.

INTRODUCTION

In some clinical studies, patient diaries are very important data to collect. It records useful information, such as questionnaire, symptom scores, and concomitant or rescue medication usage. Usually some analysis is implemented based on diary data. For example, we check and see whether a certain rescue medication is used more than 4 consecutive days, or the body pain score doubles from baseline more than 3 consecutive days. Diary data is usually huge and inconsistent, missing dates here and there. Macro CONSECUTIVEDAYS is designed to count consecutive days for diary and other similar data sets.

EXAMPLES

Suppose we have a data set, which contains a date variable. It is usually a subset from diary data satisfying the specific criteria. Here is the output of the sample data set (DIARY).

DIARY DATA SET		
Obs	SUBID	PATDYDT
1	1	25MAR2002
2	1	26MAR2002
3	1	27MAR2002
4	1	28MAR2002
5	1	30MAR2002
6	1	31MAR2002
7	1	01APR2002
8	2	26MAR2002
9	2	27MAR2002
10	2	28MAR2002

We want to find subjects with at least 4 consecutive days. We can just call the CONSECUTIVEDAYS macro. We need to specify the input data set name, subject variable, date variable, the minimum number of consecutive days needed to check, and output data set name.

After calling the macro, we will get one observation for each qualified period. The output contains subject number, number of consecutive days (CONSECUTIVE), which is equal to or greater than the given number of consecutive days, the first date of consecutive days (FIRSTDATE) and the last date of consecutive days (LASTDATE).

```
%CONSECUTIVEDAYS ( indata=Diary,
                    subject=SUBID,
                    chkdate=PATDYDT,
                    numconday=4,
                    outdata=out1)
```

OUT1 DATA SET				
Obs	SUBID	CONSECUTIVE	FIRSTDATE	LASTDATE
1	1	4	25MAR2002	28MAR2002

If we want to find the subjects with at least 3 consecutive days, then we set numconday=3.

```
%CONSECUTIVEDAYS ( indata=Diary,
                    subject=SUBID,
                    chkdate=PATDYDT,
                    numconday=3,
```

```
outdata=out2)
```

OUT2 DATA SET

Obs	SUBID	CONSECUTIVE	FIRSTDATE	LASTDATE
1	1	4	25MAR2002	28MAR2002
2	1	3	30MAR2002	01APR2002
3	2	3	26MAR2002	28MAR2002

Notice that subject 1 has two qualified records in OUT2 data set.

DESCRIPTION OF THE MACRO PROGRAM

First, we sorted incoming data by subject and date, and got rid of duplicated dates.

```
proc sort data=&indata nodupkeys;
  by &subject &chkdate;
run;
```

Then we let the first record of each subject number flag (CFLAG) equal to 1, if the second record was consecutive to the first one, we gave the same number flag as the first one and thus forwarding. When the date was not consecutive to the previous one, we added 1 to the number flag so that the consecutive dates have the same number flag. Here we used LAG function so that we could compare subjects' current record with their previous one.

```
data one;
  set &indata;
  by &subject &chkdate;
  tempdate=lag(&chkdate);
  if first.&subject then tempdate=.;
  retain cflag;
  if first.&subject then cflag=1;
  if &chkdate ne tempdate+1 then cflag=cflag+1;
run;
```

After we flagged all dates, we were able to count the number of the same flags, and output the total number of consecutive days (CONSECUTIVE) if it was greater than or equal to the given one, and the key variable flag number (CFLAG) for each subject.

```
data two(keep= &subject cflag consecutive);
  set one;
  by &subject cflag;
  retain consecutive;
  if first.cflag then consecutive=0;
  consecutive=consecutive+1;
  if last.cflag and consecutive>=&numcondays;
run;
```

By merging flagged data set ONE with data set TWO, we got data set THREE, which only had the qualified observations.

```
data three;
  merge one two(in=a);
  by &subject cflag;
  if a;
run;
```

Finally, we transformed data set (THREE) and obtained one record for each consecutive day period and output their first and last date respectively (FOUR).

```
proc sort data=three;
  by &subject cflag &chkdate;
run;

data four;
  set three;
  by &subject cflag &chkdate;
  retain firstdate;
```

```

if first.cflag then firstdate=&chkdate;
lastdate=&chkdate;
format lastdate firstdate date9.;
if last.cflag;
run;

proc sort data=four(drop=cflag &chkdate tempdate) out=&outdata;
by &subject firstdate;
run;

```

DISCUSSION

The CONSECUTIVEDAYS macro is applicable to any diary data and similar data. It is fast and easy to use. For any given diary data set, the macro tells you the number of consecutive days simply by plugging in some numbers.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Pingping Zhang
 Kendle International Inc.
 1200 Carew Tower 441 Vine Street
 Cincinnati, OH 45202
 Work Phone: 513-345-1526

APPENDIX

```

%macro CONSECUTIVEDAYS ( indata=, /* name of inputting data set */
                        subject=, /* id variable, such as patid, patno, patient */
                        chkdate=, /* date variable that we will count the number of
                                consecutive days, need to be numerical SAS date */
                        numconday=, /* at least how many consecutive days required to
                                output */
                        outdata= /* name of outputting data set */
                        );

%*****sorting data by subject and date *****;
proc sort data=&indata nodupkeys;
  by &subject &chkdate;
run;

%*****flagging the same number when date is consecutive to the previous date *****;
data one;
  set &indata;
  by &subject &chkdate;
  tempdate=lag(&chkdate);
  if first.&subject then tempdate=.;
  retain cflag;
  if first.&subject then cflag=1;
  if &chkdate ne tempdate+1 then cflag=cflag+1;
run;

%*****counting the number of same flags *****;
data two(keep= &subject cflag consecutive);
  set one;
  by &subject cflag;
  retain consecutive;
  if first.cflag then consecutive=0;
  consecutive=consecutive+1;
  if last.cflag and consecutive>=&numconday;
run;

%***** merging two (counts) with one (original) *****;
data three;
  merge one two(in=a);
  by &subject cflag;
  if a;
run;

```

```
%***** getting one observation per subject when it has at least required consecutive day ***;
proc sort data=three;
  by &subject cflag &chkdate;
run;

data four;
  set three;
  by &subject cflag &chkdate;
  retain firstdate;
  if first.cflag then firstdate=&chkdate;
  lastdate=&chkdate;
  format lastdate firstdate date9.;
  if last.cflag;
run;

proc sort data=four(drop=cflag &chkdate tempdate) out=&outdata;
  by &subject firstdate;
run;

%mend CONSECUTIVEDAYS;
```

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and ☐ other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.