

DATE AND TIME VALUES IN CLINICAL TRIAL DATA

George Li & John Morrill, both of Quintiles, Inc., Kansas City, MO

Abstract

Date and time values are widely used in clinical trial data for computing duration, cumulative time, and for windowing issues. Partial or incomplete dates in the raw data and the wide variation of formats across countries are issues that arise when working with such data. This paper discusses ways to overcome these problems using SAS® functions, formats, and other features in processing, analyzing and presenting date and time values.

Introduction

Date and time variables can be either in character form or in SAS form. With clinical trials being conducted throughout the world, a wide variety of character forms are observed. Examples of these variations, using just a few of the many possible delimiters, are shown below:

21/03/2001	03/21/2001	08:25
21MAR2001	Mar 21, 2001	08:25 am
00MAR2001	03/00/2001	08:25:00
DD/MM/2001	MM/DD/2001	15:13
DD/03/2001	03/DD/2001	03:13 pm
20010321	03212001	15:13:00

It would be cumbersome to perform computations such as sequencing, duration, or cumulation, or to present such data in a standardized form, using these character forms. Such processing and presentation is done with ease using dates and times stored in SAS form. The SAS system stores dates as the number of elapsed days since January 1, 1960, time as elapsed seconds since midnight, and composite date and time as elapsed seconds since midnight of January 1, 1960.

There are several methods of converting character date or time values into SAS form. One method is to input the variables using any of two dozen SAS date or time informats, supplemented with numerous SAS functions. A second method is to 'hard code' a constant into SAS form using the form `testdate='05NOV2001'D`; or `testtime='11:30'T`; where the D & T (or DT for composite date and time) trigger conversion of the constant into SAS

form. Once the variables are in SAS form, there are numerous formats available for presentation. See Finley (2000) and Astor (2001) for a more complete treatment of conversion and presentation methods.

These conversions to SAS form work only with complete and valid date and time values, and any departure produces a missing value. In this paper we give two examples of processing with date and time variables in SAS form and discuss various issues that arise due to missing data.

Case 1: Duration computing

Consider the following example:

```
data test;
  format subject $8. strtdate date9.
         strttime time5.
         enddate date9.  endtime time5.;
  input @1 subject $8. @10 strtdate mmdyy10.
        @21 strttime time5.
        @27 enddate mmdyy10. @38 endtime time5.;
  *** Combine date and time values ***;
  *** (results are in seconds) ***;
  start=strtdat*24*60*60+strttime;
  end   =enddate*24*60*60+endtime;
  *** Use arithmetic calculation (in hour)***;
  durat_hr=(end-start)/60/60;
  *** Use function INTCK to compute ***;
  *** duration (in hour) ***;
  intck_hr =intck('hour', start, end);
cards;
4120/001 07/15/2001 10:10 07/15/2001 11:22
4120/001 07/16/2001 09:15 07/16/2001 19:45
4120/001 07/17/2001 21:10 07/18/2001 09:20
4120/003 07/12/2001 09:59 07/12/2001 11:01
4120/003 07/13/2001          07/13/2001
4120/003          07/15/2001 11:01
4120/004 07/21/2001 10:05 07/21/2001 11:11
4120/004 07/22/2001 10:05 07/22/2001 23:11
4120/004
;
run;
proc print;
  id subject;
  by subject;
run;
```

SUBJECT	STRTDATE	STRTTIME	ENDDATE	ENDTIME	START	END	DURAT_HR	INTCK_HR
4120/001	15JUL2001	10:10	15JUL2001	11:22	1310811000	1310815320	1.2000	1
	16JUL2001	9:15	16JUL2001	19:45	1310894100	1310931900	10.5000	10
	17JUL2001	21:10	18JUL2001	9:20	1311023400	1311067200	12.1667	12
4120/003	12JUL2001	9:59	12JUL2001	11:01	1310551140	1310554860	1.0333	2
	13JUL2001	.	13JUL2001	.	.	.	.	.
	.	.	15JUL2001	11:01	.	1310814060	.	.
4120/004	21JUL2001	10:05	21JUL2001	11:11	1311329100	1311333060	1.1000	1
	22JUL2001	10:05	22JUL2001	23:11	1311415500	1311462660	13.1000	13
	.	.	.	.	.	.	.	.

With non-missing date and time values, the arithmetic calculation of duration is straightforward. Units of measurement are always important. In the example above, START and END are in seconds so a conversion factor is needed when DURAT_HR is needed in hours. Another measure of duration is produced by the INTCK function. INTCK counts the number of intervals (hour in this example) between the two date-time values and its results can easily be misinterpreted. Subject 4120/003's first observation, for example, started at 9:59 and ended at 11:01. INTCK returns 2 hours since there are 2 hour interval (10 & 11) between the values, but the duration is really just slightly over an hour.

Note that if any part of the duration computation is missing, both measures of duration produce a missing observation. Knowing why duration is missing is important. If duration is measured in years and a time is all that is missing, a conservative assumption may provide a suitable result and keep the observation from being missing. This will be discussed further after the second case.

Incidentally, it is worthwhile to pay attention to precision issues. As DURAT_HR above can in reality be a repeating decimal, comparisons made using this may produce different results depending on the precision of the host. One suggestion to avoid such problems is to make comparisons between variables expressed in integers. In this case, expressing duration in seconds would be the appropriate thing to do.

way to compute hours on treatment for each subject would be to transpose the data to make one observation per subject, and then create a cumulative variable by adding together each duration. Of course if any duration is missing, the cumulation of them will also be missing using this method. A variation would be to use the SUM function in the creation of the new cumulative variable. The SUM function treats missing values as zero and so produces a non-missing result. Depending on your objective, this feature can be both convenient and dangerous.

A way to compute cumulative time without transposing the data is shown in the following code:

```
data test;
  set test;
  * Continuing with the data from above *;
  by subject;
  *** hours is a summary variable ***;
  *** to accumulate durat_hr values ***;
  hours+durat_hr;

  *** hours is assigned at first observation ***;
  *** for each subject and then it will be ***;
  *** retained as defined above ***;
  if first.subject then hours=durat_hr;
  *** The cumulative value is obtained ***;
  *** on the last observation for a subject ***;
  if last.subject then tot_hours=hours;
  drop hours;
run;
proc print;
  id subject;
  by subject;
  var startdate starttime enddate endtime
      durat_hr tot_hours;
run;
```

Case 2: Cumulative time

Analysis of clinical data often involves computing cumulative time such as total duration of treatment. One

subject	startdate	starttime	enddate	endtime	durat_hr	tot_hours
4120/001	15AUG2001	10:10	15AUG2001	11:22	1.2000	.
	16AUG2001	9:15	16AUG2001	19:45	10.5000	.
	17AUG2001	21:10	18AUG2001	9:20	12.1667	23.8667
4120/003	12AUG2001	9:59	12AUG2001	11:01	1.0333	.
	13AUG2001	.	13AUG2001	.	.	.
	.	.	15AUG2001	11:01	.	1.0333
4120/004	21AUG2001	10:05	21AUG2001	11:11	1.1000	.
	22AUG2001	10:05	22AUG2001	23:11	13.1000	.
	.	.	.	.	.	14.2000

We note three things about the sum statement in which DURAT_HR is accumulated into the HOURS variable. First, there is an implicit HOURS= to the left of the HOURS+DURAT_HR; statement. Second, this sum statement works even for missing values of DURAT_HR because it ignores the missing values. Third, HOURS is retained even though there is no explicit RETAIN statement.

On being conservative

If we require an assumption regarding missing dates, it is prudent to take a conservative approach in supplying the replacement date. If we are computing exposure to study medication and the start dose day is missing, the conservative approach may be to assume the missing day is the *first* of the month. But we would not choose the *first* month, January, to replace a missing start-of-disease month if we are computing disease duration and X months of some disease is required for inclusion into a study.

Furthermore, we should consider other information in supplying this replacement date. For example, it may be appropriate to use the date of randomization or informed consent as the start of study medication rather than blindly use the first of the month. As another example, let us assume that a study begins with a placebo lead-in followed by an active drug. We may be tempted to assign the first day of the month to the missing day of an Adverse Event, but in this case we risk not counting a potential Treatment Emergent Adverse Event. So, we must be careful how we define conservative!

Avoid problems before they are problems

Thinking ahead, including proper case report form (CRF) design, can often avoid date and time problems. We now present two cases that illustrate this. First, consider a CRF diary designed with the intention of collecting wake-up and bedtime patient-reported symptoms. Fields exist for the date, morning time and score, and evening time and score. What happens when the subject records a bedtime score after midnight as shown in the second observation below?

Date	wake-up		bedtime	
	time	score	time	score
01AUG2001	06:45	3	22:15	2
02AUG2001	06:30	3	00:15	4
03AUG2001	07:15	5	22:00	2

The date associated with the second row inappropriately places the bedtime score 24 hours before it actually happens, just two hours after the 01AUG2001 bedtime score. This issue also arises in pharmacokinetic studies with hourly blood draws. If one date is recorded per page,

the last few blood draws which actually occur past midnight appear in the resulting database as occurring *before* the first blood draw. These difficulties can be avoided by requiring each time to be associated with its own date.

In the example above, what if the bedtime score time on the second record was incorrectly entered as 12:15? The date would be correct but the score may not fall within a protocol-specified bedtime window and would thus be counted as missing. This has further implications if imputation is involved. It should go without saying that all times should be entered in 24-hour format.

Second, consider a study which requires a score to be taken immediately prior to a medication and that times are recorded for both events for verification of the order. If multiple, non-synchronized clocks are used, several false violations may be flagged. There are several ways of avoiding this. One would be to use the same clock, but even this may produce identical times for both events (if only hours and minutes are collected) so that order cannot be established. Another would be to replace the collection of the score's time with the question, "Was the score taken within Y minutes before the medicine?" Another requires an understanding of how fast the medicine takes effect: establish a window of time that would allow the medicine to be taken W minutes before the score and not be counted as a violation. Depending on how far apart the multiple clocks are, this in itself may eliminate the false violators.

For another example of this second issue, but with a twist, consider that some studies use devices that record scores and transmit them electronically to a central location. These scores, along with a date and time stamp internal to the device, become part of the study database. Assume the protocol requires a subject to use such a device and then immediately take a medication, recording the time and date of medication on the CRF. Since the clock used by the subject is not the same clock used by the device, some variation would be natural to expect and the window solution described in the previous paragraph could be used. But the twist is that the device is not designed to take into account daylight savings time for the region of the country in which it is used. In this case, a subject may consistently take the score at 08:00 and the medicine at 08:01 throughout the study. Assuming the clocks are synchronized, the databases would reflect this correctly until a daylight savings time change. Then, the device records 09:00 when the time is really 08:00 and the subject falsely appears as a violator. Unless the window W is 60 minutes, the lesson here is to use a device manufactured to show its internal time externally.

Conclusion

Partial date and time values in raw data present issues that must be considered carefully when working with clinical data. We have brought up some of these issues and discussed ways to arrive at a solution.

References

Astor, Rick, SAS[®] Routines for Calendar and Clock Elements, presented at PharmaSUG 2001 (Boston, MA, May 2001) and published in the Proceedings of PharmaSUG 2001.

Finley, Wayne, A Beginners Guide to SAS[®] Date and Time Handling, presented at the 25th annual SAS User's Group International, SUGI, meetings (Indianapolis, IN, April 2000) and published in the Proceedings of the 25th Annual SUGI Conference, 2001. This paper may be downloaded from www2.sas.com/proceedings/sugi25/PROCCED.pdf.

Author Contact

Quintiles, Inc.
P.O. Box 9708
Kansas City, MO 64134-0708

George.Li@Quintiles.com
John.Morrill@Quintiles.com