

Adjusting for daylight saving times

PhUSE Frankfurt, 06Nov2018, Paper CT14

Guido Wendland

1. The problem

Introduction: DST (Daylight saving times) around the world

Country	Regions/States	DST Start Date	DST End Date
Belgium	All locations	Sunday, March 25	Sunday, October 28
Japan	All locations	No DST in 2018	
New Zealand	All locations	Sunday, September 30	Sunday, April 1
United States	Most locations	Sunday, March 11	Sunday, November 4
	Hawaii, most of Arizona	No DST in 2018	

2nd Sunday
in March



1st Sunday
in Nov.



DST=
Daylight-saving
time



SST=
Standard Time

SST=
Standard Time



Last Sunday
in March



Last Sunday
in Oct.

Example nonmem file

Combination of various data

- Dosing
- Pharmacokinetic
- Efficacy

Time since first/last dose

+	P	Q	R	S	T
SMDT	PCDT	TIME	SCHT	TSFD	TSLD
2010-10-26T08:14	.	0	-100	-217.933	.
2010-11-04T08:17	2010-11-04T08:20	217.05	0	-0.883	-0.883
2010-11-04T08:20	.	217.1	0	-0.833	-0.833
2010-11-04T09:10	.	217.933	.	0	0
2010-11-04T09:40	.	218.433	0.5	0.5	0.5
2010-11-04T10:10	.	218.933	1	1	1
2010-11-04T11:06	2010-11-04T11:10	219.867	2	1.933	1.933
2010-11-04T11:10	.	219.933	2	2	2
2010-11-04T12:10	.	220.933	3	3	3
2010-11-04T13:06	2010-11-04T13:10	221.867	4	3.933	3.933
2010-11-04T13:10	.	221.933	4	4	4
2010-11-04T15:10	.	223.933	6	6	6
2010-11-04T17:05	2010-11-04T17:10	225.85	8	7.917	7.917
2010-11-04T17:10	.	225.933	8	8	8
2010-11-04T19:10	.	227.933	10	10	10
2010-11-04T21:06	2010-11-04T21:10	229.867	12	11.933	11.933
2010-11-04T21:10	.	229.933	12	12	12
2010-11-05T00:10	.	232.933	15	15	15
2010-11-05T09:06	2010-11-05T09:10	241.867	24	23.933	23.933
2010-11-05T09:10	.	241.933	24	24	24
2010-11-05T21:10	.	253.933	36	36	36
2010-11-06T09:06	2010-11-06T09:10	265.867	48	47.933	47.933
2010-11-06T09:10	.	265.933	48	48	48
2010-11-06T21:10	.	277.933	60	60	60
2010-11-07T09:06	2010-11-07T09:10	289.867	72	71.933	71.933

2. The macro `%_dst`

The macro %_dst

- Are datetimes during daylight saving time?

- Factors:

- Country or region
- Year



Covered by (standard)
reference dataset _dst

- Input data:

- Datetime variables (character vs numeric)
- Single or multiple dates to be considered.

Standard reference dataset _dst

Region	Country Codes (according to ISO 3166)	Start and End of daylight saving time
E(U)	AX, AL, AD, BE, BA, BG, DK, DE, EE, FO, FI, SF, FR, GI, GR, GL, GG, IM, IE, IT, JE, XK, HR, LV, LB, LI, LT, LU, MT, MA, MK, MD, MC, ME, NL, NO, AT, PL, PT, RO, SM, SE, CH, RS, SK, SI, ES, CZ, UA, HU, VA, UK, EH, CY	last Sunday of March until last Sunday of October
U(S)	BS, BM, HT, CA, CU, TC, US	2 nd Sunday of March until 2 nd Sunday of November
N	AF, EG, DZ, AS, VI, AO, AI, AG, AR, AM, AW, AC, AZ, ET, BH, BD, BB, BZ, BJ, BT, BO, BW, BV, VG, IO, BN, BF, BI, CN, CP, CK, CR, CW, DM,	No daylight-saving time

_dst  Start  End Start  End	_year	start1	end1	start2	end2
	2016	2016-03-13T02:00:00	2016-11-06T02:00:00	2016-03-27T02:00:00	2016-10-30T02:00:00
	2017	2017-03-12T02:00:00	2017-11-05T02:00:00	2017-03-26T02:00:00	2017-10-29T02:00:00
	2018	2018-03-11T02:00:00	2018-11-04T02:00:00	2018-03-25T02:00:00	2018-10-28T02:00:00
	2019	2019-03-10T02:00:00	2019-11-03T02:00:00	2019-03-31T02:00:00	2019-10-27T02:00:00

Parameters of %_dst macro: Input: Multiple regions

Input

ds = data1

var = dt1dtc dt2n

refds = dst (default)

region=

%_dst

dsout = dsx1

data1

usubjid	dt1dtc	dt2n
KF101-DE1001001	2017-03-25T03:25	1806024300
KF101-US2001001	2017-03-25T01:25	1806031500
KF101-DE1002001	2018-03-25T03:25	1837560300
KF101-US2001002	2018-03-25T01:25	1837567500

=2017-03-25T01:25
=2017-03-25T03:25
=2018-03-25T01:25
=2018-03-25T03:25

Default reference dataset _dst is used

Region determined from variable USUBJID

ISO8601 datetime format (character)

Numeric datetime



Start 

2017-03-12T02:00:00

2018-03-11T02:00:00



Start 

2017-03-26T02:00:00

2018-03-25T02:00:00

Parameters of %_dst macro: Output, Region from USUBJID

ds = data2
var = dt1dtc dt2n
refds = _dst (default)
region=

%_dst

Output

dsout = dsx2



Start 

2017-03-12T02:00:00
2018-03-11T02:00:00



Start 

2017-03-26T02:00:00
2018-03-25T02:00:00

Derived output information



data2 dsx2

USUBJID	DT1DTC	DT2N	DT1DTC_DST	DT2N_DST	REGION	DT1DTC_DSTN	DT2N_DSTN
KF101-DE-001001	2017-03-25T03:25	1806024300	SST	SST	E	0	0
KF101-US-001001	2017-03-25T01:25	1806031500	DST	DST	U	-1	-1
KF101-DE-002001	2018-03-26T03:25	1837560300	DST	SST	E	-1	0
KF101-US-001002	2018-03-26T01:25	1837567500	DST	DST	U	-1	-1

2. Coding solutions

1. Generation of reference dataset _dst: Europe vs USA



First day of March (x1): `x1=mdy(3,1,year);` e.g. 01/03/2018
Increment for 2nd Sunday (y1): `y1=(weekday(x1) ne 1) +1;` Sunday is weekday 1!
Date->Datetime + Formatting: `z1=put(86400*intnx("week",x1,y1)+7200,E8601DT.);`

2. Determine variables and variables types in input dataset

- Select data set (DS) (with &LIBNAME and &MEMNAME.)
- Select variables (VAR) under investigation
- Determine variables and corresponding variables types

```
data _vcolumn;  
    set sashelp.vcolumn (where=(upcase(libname)="%upcase(&LIBNAME.)"  
and upcase(memname)="%upcase(&MEMNAME.)" and upcase(name) in  
        %f_list(list=%upcase(&VAR.),quotes=Y))));  
    keep libname memname name type;  
run;
```

_vcolumn

Library Name	Member Name	Column Name	Column Type
WORK	DS2	dt1dtc	char
WORK	DS2	dt2n	num

2. Determine variables and variables types in input dataset

- Select data set (DS) (with &LIBNAME and &MEMNAME.)
- Select variables (VAR) under investigation
- **Generate list of variables and variable types**

```
proc sql noprint;  
    select name, type  
    into : _vars separated by " ",  
         : _vartypes separated by " "  
    from _vcolumn;  
quit;
```

**Local macro variables:
_vars and _vartypes**

`_VARS=dt1dtc dt2n, _VARTYPES=char num`

3. Determine reference year + region for each variable and record.

- Determine the reference year for all datetime variables and each record

Character variable: `_yearref1=input(substr(dt1dtc,1,4),best.);`

Numeric variable `_yearref2=year(datepart(dt2n));`

- Determine the region (i.e. the algorithm for deriving start and end of daylight saving time)

— Provide region centrally

— Derive the region from e.g. the subject id

Region	Country Codes (according to ISO 3166)	Start and End of daylight saving time
E(U)	..., DE,...	last Sunday of March until last Sunday of October
U(S)	..., US,...	2 nd Sunday of March until 2 nd Sunday of November

data2 ⇒ **_refdate01**

usubjid	dt1dtc	dt2n	_yearref1	dt1dtcc	_yearref2	dt2nc	region
KF101-DE1001001	2017-03-25T03:25	1806024300	2017	2017-03-25T03:25	2017	2017-03-25T01:25	E
KF101-US2001001	2017-03-25T01:25	1806031500	2017	2017-03-25T01:25	2017	2017-03-25T03:25	U
KF101-DE1002001	2018-03-26T03:25	1837560300	2018	2018-03-26T03:25	2018	2018-03-25T01:25	E
KF101-US2001002	2018-03-26T01:25	1837567500	2018	2018-03-26T01:25	2018	2018-03-25T03:25	U

4. Determine reference year for each variable and record.

- Merge the reference data set (_dst) iteratively by reference year to retrieve the dst time for each record.

```
%let _cnt=1;
%do %while(%qscan(&_vars., &_cnt, %str( )) ne %str( ));
    %let _cnt1=%eval(&_cnt.+1);
    proc sql;
        create table _refdate0&_cnt1. as
            select a.*, %do _i=1 %to &_nobs1; start&_i. as start&_i._&_cnt.,
                                                    end&_i. as end&_i._&_cnt.,
                %end;
                b._year as _year_&_cnt.
            from _refdate0&_cnt. a left join &REFDS. b
            on a._yearref&_cnt.=b._year;
    quit;
    %let _cnt=%eval(&_cnt+1);
%end;
```

Loop through the different variables to check for DST

4. Determine reference year for each variable and record.

- Merge the reference data set (_dst) iteratively by reference year to retrieve the dst time for each record.

```
%let _cnt=1;
%do %while(%qscan(&_vars.,&_cnt,%str( )) ne %str( ));
  %let _cnt1=%eval(&_cnt.+1);
  proc sql;
    create table _refdate0&_cnt1. as
      select a.*,%do _i=1 %to &_nobs1; start&_i. as start&_i.&_cnt.,
                                                end&_i. as end&_i.&_cnt.,
      %end;
      b._year as _year_&_cnt.
    from _refdate0&_cnt. a left join &REFDS. b
    on a._yearref&_cnt.=b._year;
  quit;
  %let _cnt=%eval(&_cnt+1);
%end;
```

Use proc sql iteratively to add start and end times for each variable

4. Determine reference year for each variable and record.

- Merge the reference data set (_dst) iteratively by reference year to retrieve the dst time for each record.

```
%let _cnt=1;
%do %while(%qscan(&_vars.,&_cnt,%str( )) ne %str( ));
  %let _cnt1=%eval(&_cnt.+1);
  proc sql;
    create table _refdate0&_cnt1. as
      select a.*,%do _i=1 %to &_nobs1; start&_i. as start&_i._&_cnt.,
                                             end&_i. as end&_i._&_cnt.,
                                             %end;
                                             b. year as year & cnt.
    from _refdate0&_cnt. a left join &REFDS. b
    on a._yearref&_cnt.=b._year;
  quit;
  %let _cnt=%eval(&_cnt+1);
%end;
```

Usually
_dst

Map input data (_refdate0&_cnt) and reference data (_dst) by year

4. Determine reference year for each variable and record.

- Merge the reference data set (_dst) iteratively by reference year to retrieve the dst time for each record.

```
%let _cnt=1;
%do %while(%qscan(&_vars.,&_cnt,%str( )) ne %str( ));
  %let _cnt1=%eval(&_cnt.+1);
  proc sql;
    create table _refdate0&_cnt1. as
      select a.*,%do _i=1 %to &_nobs1; start&_i. as start&_i._&_cnt.,
                                     end&_i. as end&_i._&_cnt.,
      %end;
      b._year as _year_&_cnt.
    from _refdate0&_cnt. a left join &REFDS. b
    on a._yearref&_cnt.=b._year;
  quit;
  %let _cnt=%eval(&_cnt+1);
%end;
```

Generate a pair of start+end variables for each region

(&_nobs1=Number of regions)

5. Comparison and output generation

ISO8601 dates are used for date comparison (alphabetic ordering).

data1 **dsx1**

	DT1DTC	DT2N	DT1DTC_DST	DT2N_DST	DT1DTC_DSTN	DT2N_DSTN
1	2017-03-25T03:25	1806024300	SST	SST	0	0
2	2017-03-25T01:25	1806031500	SST	SST	0	0
3	2018-03-25T03:25	1837560300	DST	SST	-1	0
4	2018-03-25T01:25	1837567500	WIT	DST	0	-1

data2 **dsx2**

	USUBJID	DT1DTC	DT2N	DT1DTC_DST	DT2N_DST	REGION	DT1DTC_DSTN	DT2N_DSTN
KF101	DE1001001	2017-03-25T03:25	1806024300	SST	SST	E	0	0
KF101	US2001001	2017-03-25T01:25	1806031500	DST	DST	U	-1	-1
KF101	DE1002001	2018-03-26T03:25	1837560300	DST	SST	E	-1	0
KF101	US2001002	2018-03-26T01:25	1837567500	DST	DST	U	-1	-1

4. Points to consider

Other issues

- Additional or unknown regions
 - Example Arizona
 - Problematic if there are different daylight saving times in one country
- Post-processing
 - Use *_DSTN variable to adjust time differences between two timepoints.
- Use of different reference dataset required variable “region0” in the input data file
- Changes of DST over time

