MWSUG 2017 – Paper RF05

Macro that can Provide More Information for your Character Variables

Ting Sa, Cincinnati Children's Hospital Medical Center, Cincinnati, OH

ABSTRACT

Sometimes, we want to change character variables to numerical variables in batches. But before doing that, we may need to manually check if those variables contain only numerical values. Also sometimes, we want to make all the date or datetime formats be consistent across the SAS® data sets, but if those variables are saved as character variables and we don't have a data dictionary, we will have to manually check the data sets to see their formats. Besides, we may have those character variables that only contain blank values and we can delete them to save space. Using the macro in this paper, you can get those information for each character variable without manually checking one by one. The macro can check all the character variables in a library or in some data sets. An excel report will be generated after running the macro to include the information about each checked character variable. Users can use the excel report to further filter the information. Based on the information, the user can decide the things they want to do for the character variables, like if the character variables contain all numerical values, they can be transformed to a numeric variable, if the character variables contain missing values, they can delete those variables etc.

INTRODUCTION

Sometimes, we want to know more information about a character variable, like if the variable only contains numeric values, or contains no values or contains date and datetime values. Based on the information we get, we can do things in batches, like convert character values to numeric values, delete those character variables with no values, make date and datetime variables be consistent across the data sets. However, to get those information, we need to either check the data dictionary or need to manually open the data sets to check. Using the macro in this paper, it can provide you with the information automatically. All the macro codes will be included in this paper. Besides, I will show you how to use the macro.

THE SAMPLE DATA SET

Running the following SAS codes, you will get two SAS data sets "sample1" and "sample2". Figure 1 and Figure 2 contain the screenshots of these two SAS data sets.

```
data sample1;
length a b c $100.;
do d=1 to 10;
if mod(d,3)=0 then a=d;
else if mod(d,3)=1 then a=cats("+",d);
else if mod(d,3)=2 then a=cats("-",d);
if mod(d,3)=0 then b=cats(d,".","12345");
else if mod(d,3)=1 then b=cats("+",d,".","12345");
else if mod(d,3)=2 then b=cats("-",d,".","12345");
c="this is just test";
output;
end;
run;
data sample2;
format e yymmdd10.;
length f g h i j $100.;
do k=1 to 10;
e="01Oct2016"d+k;
f=put(e,date9.);
```

```
g=put(e,mmddyy10.);
h=put(e,yymmdd10.);
i=cats(put(e,date9.),":00:00:00");
j=cats(put(e,mmddyy10.),":18:59");
output;
end;
run;
```

```
data sample2;
set sample2;
if _N_=2 then call missing(of e--j);
run;
```

	a	b	c	d
1	+1	+1.12345	this is just test	1
2	-2	-2.12345	this is just test	2
3	3	3.12345	this is just test	3
4	+4	+4.12345	this is just test	4
5	-5	-5.12345	this is just test	5
6	6	6.12345	this is just test	6
7	+7	+7.12345	this is just test	7
8	-8	-8.12345	this is just test	8
9	9	9.12345	this is just test	9
10	+10	+10.12345	this is just test	10

Figure 1 the Screenshot for the Sample1 Data Set

	e	f	g	h	i	j	k
1	2016-10-02	02OCT2016	10/02/2016	2016-10-02	02OCT2016:00:00:00	10/02/2016:18:59	1
2	.						2
3	2016-10-04	04OCT2016	10/04/2016	2016-10-04	04OCT2016:00:00:00	10/04/2016:18:59	3
4	2016-10-05	05OCT2016	10/05/2016	2016-10-05	05OCT2016:00:00:00	10/05/2016:18:59	4
5	2016-10-06	06OCT2016	10/06/2016	2016-10-06	06OCT2016:00:00:00	10/06/2016:18:59	5
6	2016-10-07	07OCT2016	10/07/2016	2016-10-07	07OCT2016:00:00:00	10/07/2016:18:59	6
7	2016-10-08	08OCT2016	10/08/2016	2016-10-08	08OCT2016:00:00:00	10/08/2016:18:59	7
8	2016-10-09	09OCT2016	10/09/2016	2016-10-09	09OCT2016:00:00:00	10/09/2016:18:59	8
9	2016-10-10	10OCT2016	10/10/2016	2016-10-10	10OCT2016:00:00:00	10/10/2016:18:59	9
10	2016-10-11	11OCT2016	10/11/2016	2016-10-11	11OCT2016:00:00:00	10/11/2016:18:59	10

Figure 2 the Screenshot for the Sample2 Data Set

Figure 3 shows the data types for the variables in “sample1” and “sampl2” data sets. We can see that variables “a”,”b”,”c”,”f”,”g”,”h”,”i”,”j” are character variables, but we don’t know further information of those variables, like what kind of values are contained in those variables. I am going to use these 2 sample data sets to show you how the macro works.

	Library Name	Member Name	Column Name	Column Number in Table	Column Type
1	WORK	SAMPLE1	a	1	char
2	WORK	SAMPLE1	b	2	char
3	WORK	SAMPLE1	c	3	char
4	WORK	SAMPLE1	d	4	num
5	WORK	SAMPLE2	e	1	num
6	WORK	SAMPLE2	f	2	char
7	WORK	SAMPLE2	g	3	char
8	WORK	SAMPLE2	h	4	char
9	WORK	SAMPLE2	i	5	char
10	WORK	SAMPLE2	j	6	char
11	WORK	SAMPLE2	k	7	num

Figure 3 the Data Type Informaton for the Sample1 and Sample2 Data Sets

THE CHECKCHARVARTYPE MACRO

Presented below are the SAS codes for the macro:

```
%macro checkCharVarType(libname=,datasets=%str(),filedir=);
%macro checkcharvars(libnm=,datanm=,varnm=,tno=);
proc sql;
create table tchar_&tno. as
select distinct "&libnm." as libnm length=32,
"&datanm." as datanm length=32,
"&varnm." as varnm length=32,
&varnm. as value length=1000
from &libnm..&datanm.(keep=&varnm.);
quit;
%mend;

%let datasets=%sysfunc(upcase(&datasets.));

proc sql noprint;
*select all the char variables from the data sets and save them in data set
tmp1;
create table tmp1 as
select libname,memname,memtype,name
from dictionary.columns
where libname=upcase("&libname.") and memtype="DATA" and type="char"
%if &datasets.^= %then %do;
and upcase(memname) in (&datasets.)
%end;
;
quit;

data tmp2;
length libnm datanm varnm $32.;
set tmp1;
keep libnm datanm varnm;
libnm=libname;
datanm=memname;
varnm=name;
attrib _all_ label='';
run;
proc sort data=tmp2;by libnm datanm varnm;run;
```

```
data tmp2;
set tmp2;
length sascodes $500.;
sascodes=cats('%checkcharvars(libnm=',libnm,',datanm=',datanm,',varnm=',varnm
,',tno=',_n_,');');
run;
data _null_;
set tmp2;
call execute(sascodes);
run;

data tmp3;
set tchar_:;
run;
proc sort data=tmp3;by value;run;

data tmp4;
set tmp3;
length chartype $50.;
if value="" then chartype="missing";
else do;
value1=compress(value,'0123456789');
if value1 in ("") or (substr(value,1,1) in ("+","-") and value1 in ("+","-"))
then chartype="integer";
else if value1 in (".") or (substr(value,1,1) in ("+","-") and value1 in
("+.","-.")) then chartype="decimal";
else if value1="--" and length(value) in (8,10) then chartype="date_hyphen";
else if value1="//" and length(value) in (8,10) then chartype="date_slash";
else if lowcase(value1) in ('jan','feb','mar','apr','may','jun',
                            'jul','aug','sep','oct','nov','dec') and
length(value) in (7,9) then chartype="date_dateM";
else if value1 in ("--::","--:::") then chartype="datetime_hyphen";
else if value1 in ("//::","//:::") then chartype="datetime_slash";
else if lowcase(value1) in ('jan::','feb::','mar::','apr::','may::','jun::',
                            'jul::','aug::','sep::','oct::','nov::','dec::',
                            'jan:::','feb:::','mar:::','apr:::','may:::',
     'jun:::', 'jul:::','aug:::','sep:::','oct:::',        'nov:::','dec:::')
then chartype="datetime_dateM";
else chartype="undefined";
end;
keep libnm datanm varnm value value1 chartype;
run;
proc sort data=tmp4 out=tmp5 nodupkey;by libnm datanm varnm chartype;run;
proc transpose data=tmp5 out=tmp5(drop=_name_) prefix=type;by libnm datanm
varnm;var chartype;run;
data tmp5;
length libnm datanm varnm $32.;
set tmp5;
alltype=catx(";",of type:);
drop type:;
run;

data tmp1;
set tmp1;
rowno=_n_;
rename name=varnm;
run;
```

```
proc sort data=tmp1;by varnm;run;
proc sort data=tmp5;by varnm;run;

data tmp1;
merge tmp1 tmp5;
by varnm;
if chartype="" then chartype="missing";
label libname="library" memname="Data Name" varnm="Variable Name"
alltype="Character Info";
run;
proc sort data=tmp1;by rowno;run;

PROC EXPORT DATA= tmp1(drop=memtype rowno libnm datanm chartype)
        OUTFILE= "&filedir."
        DBMS=EXCEL REPLACE label;
      SHEET="charinfo";
RUN;
proc datasets noprint;delete tchar: tmp:;run;quit;
%mend;
```

- The “libname” is used to indicate the library name for the input dataset.
- The “datasets” is used to indicate the input SAS dataset names. If you don’t pass a value to it, the macro will check all the data sets in the library.
- The “filedir” is used to indicate the file location you want to save the excel report.

For the “sample1” and “sample2” test data sets in this paper, we can call the macro using the following way and it will create an excel report “c:\charinfo.xlsx”.

```
%checkCharVarType(libname=work,datasets=%str('sample1','sample2'),
filedir=%str(C:\charinfo.xlsx));
```

Figure4 is the screenshot for the “c:\charinfo.xlsx” report.

	A	B	C	D
1	library	Data Name	Variable Name	Character Info
2	WORK	SAMPLE1	a	integer
3	WORK	SAMPLE1	b	decimal
4	WORK	SAMPLE1	c	undefined
5	WORK	SAMPLE2	f	date_dateM;missing
6	WORK	SAMPLE2	g	date_slash;missing
7	WORK	SAMPLE2	h	date_hyphen;missing
8	WORK	SAMPLE2	i	datetime_dateM;missing
9	WORK	SAMPLE2	j	datetime_slash;missing

Figure 4 the Screenshot for the “c:\charinfo.xlsx” Report

WHAT THE VALUES THE MACRO CHECKS

Currently, the macro will check if a character variable contains those type of values:

(1) The macro can find out if a variable contains integers or not, even the integers contain “+” and “-“in front of the values.

(2) The macro can find out if a variable contains decimal values or not, even the decimal values contain "+" and "-"in front of the values.

(3) The macro can find out if a variable contains any missing values.

(4) The macro can find values that only contain digits and two "/", this helps to identify those date values in mmddyy formats.

(5) The macro can find values that only contain digits and two "--", this helps to identify those date values in yymmdd formats.

(6) The macro can find values that only contain digits and character values in this list: ('jan','feb','mar','apr','may','jun', 'jul','aug','sep','oct','nov','dec'), this helps to identify those date values in dateM formats.

(7) The macro can find values that only contain digits,":" and two "--", this helps to identify those datetime values where date is in yymmdd formats.

(8) The macro can find values that only contain digits, ":" and character values in this list: ('jan','feb','mar','apr','may','jun', 'jul','aug','sep','oct','nov','dec'), this helps to identify those datetime values where date is in dateM formats.

Below are the SAS codes that is used to check the variable values, you could always update this part to either make the checking more accurate or adding other checking as well:

```
if value="" then chartype="missing";
else do;
value1=compress(value,'0123456789');
if value1 in ("") or (substr(value,1,1) in ("+","-") and value1 in ("+","-"))
then chartype="integer";
else if value1 in (".") or (substr(value,1,1) in ("+","-") and value1 in
("+.","-.")) then chartype="decimal";
else if value1="--" and length(value) in (8,10) then chartype="date_hyphen";
else if value1="//" and length(value) in (8,10) then chartype="date_slash";
else if lowcase(value1) in ('jan','feb','mar','apr','may','jun',
                                'jul','aug','sep','oct','nov','dec') and
length(value) in (7,9) then chartype="date_dateM";
else if value1 in ("--::","--:::") then chartype="datetime_hyphen";
else if value1 in ("//::","//:::") then chartype="datetime_slash";
else if lowcase(value1) in ('jan::','feb::','mar::','apr::','may::','jun::',
                            'jul::','aug::','sep::','oct::','nov::','dec::',
                            'jan:::','feb:::','mar:::','apr:::','may:::',
     'jun:::', 'jul:::','aug:::','sep:::','oct:::',        'nov:::','dec:::')
then chartype="datetime_dateM";
else chartype="undefined";
end;
```

The macro will check all those type of values in the selected character variables, if those character variables contain those type of values, the macro will provide this information to the user.

Figure 5 shows you the description for each value:

Value	Description
missing	contain missing values
integer	contain integer values
decimal	contain decimal values
date_hyphen	may contain date values with -
date_slash	may contain date values with /

date_dateM	may contain date values in dateM. format
datetime_hyphen	may contain datetime values with -
datetime_slash	may contain datetime values with /
datetime_dateM	may contain datetime values with dateM. format
missing value	may contain missing values
not in those defined type of character values	may contain characters values

Figure 5 the List of the Character Info Value Description

So if a variable contains missing values and integer values, the description for this variable in the final excel report will be “integer;missing”.

Though the macro currently cannot be 100% accurate to check if a value is a date value or a datetime value, but it works in most of the cases.

Besides, the macro will be helpful to those users who want to change character variables to numeric variables because the macro can identify the integer and decimal values correctly. Also the macro is helpful to the users who want to delete those character variables that don’t contain any values in batches.

Also the macro is working efficiently, for most of the clinical trial study, it runs fast and will give you the results in about 2 or 3 minutes at most. I just use the base SAS to run this macro. If you have a SAS server, it will be even quicker.

CONCLUSION

The macro presented in this paper can be used as a helpful tool to provide more character variable information to the users. The users can further extend this macro function by adding checking to the macro.

ACKNOWLEDGMENTS

The author wishes to thank the Division of Biostatistics and Epidemiology at Cincinnati Children's Hospital Medical Center for its support.

CONTACT INFORMATION

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

Ting Sa
Division of Biostatistics and Epidemiology, Cincinnati Children’s Hospital Medical Center
513-636-3674
Ting.Sa@CCHMC.ORG