```
%macro md_byvar_check(in_data=
                     ,byvars=
                     ,byvars_preloadfmt=
                     ,byvars_sort_order=
                     ,byvars_sort_asc_or_desc=) /store source des='V1.0.0.3';
%*************************************************************************
*********************
* FILENAME:    MD_BYVAR_CHECK.SAS
* DEVELOPER:   (b) (6)
* PLATFORM:    SAS 9.1.3, 9.2 on PC
*
* MACROS USED: None
* ASSUMPTIONS:
*
*
* DESCRIPTION:
*   This macro will be called in calling program to extract and verify the BYVARS
information from input macro parameter. It will return:
*    1. byvar1 - byvar&num_byvars : extracted from BYVARS
*    2. fmt1   - fmt&num_fmts : extracted from BYVARS_PRELOADFMT
*    3. sort1  - sort&num_sorts : extracted from BYVARS_SORT_ORDER
*    4. asc_or_desc1 - asc_or_desc&num_asc_or_desc: extracted from
BYVARS_SORT_ASC_OR_DESC
*   It also return:
*    5. byvar_start1 - byvar_start&num_byvars  if BYVARS_PRELOADFMT is not blank
*    6. byvar_notsorted1_nobs - byvar_notsorted&num_byvars._nobs if
BYVARS_PRELOADFMT is not blank
*
*
* PARAMETERS:
* BYVARS = by variable to obtain counts (typically LBTESTCD VISIT)- space
delimited.
*    (default = Blank) (OPTIONAL)
*
* BYVARS_PRELOADFMT =  the formats associated with BYVARS to preload in PROC MEANS
- space delimited.
*        If do not want a format with a respective BYVARS then use the indicator
word BLANK.  For example,
*        if BYVARS = LBTESTCD VISIT and there is no preload format associated with
BYVARS but there is a preload
*        format associaged with VISIT, called VISFMT, then call would be:
*
*                        ,BYVARS = LBTESTCD VISIT
*                        ,BYVARS_PRELOADFMT = BLANK VISFMT
*
*    If format is BLANK, the BYVAR will be used in the BY statement rather than a
CLASS statement in PROC MEANS.
*        This format serves two main functions:
*
*    1) it preloads, meaning that zero counts can be generated even if there are no
```

```
subjects that
*    meet the criterion for a particular row.
*
*    2) The format is used for display purposes.
*
*        Multilabel formats are supported but user should ascertain results are as
expected.
*
*    (default = blank) (OPTIONAL)
*
* BYVARS_SORT_ORDER = Value should be I, F, N.
*    I = sort by internal value of BY variable.
*    F = sort by formatted value of BY variable.
*    N = NOTSORTED option used when creating format specified in BYVARS_PRELOADFMT
- BY variable will be sorted in same order
*        as when format was created.
*
*    If parameter is F, but PRELOADFMT is BLANK, then check dataset for attached
format. If not exist then defaults to I - an alert will be
*    issued.
*
*    If parameter is N, but PRELOADFMT is BLANK, then check dataset for attached
format. If not exist then defaults to I - an alert will be
*    issued.  If PRELOADFMT or attached format was created without using the
NOTSORTED option, then an alert
*    will be issued and parameter will default to I.
*
*    (default = I) (OPTIONAL)
*
* BYVARS_SORT_ASC_OR_DESC = (D)escending or (A)scending, specify in which way the
*    BY variable should be sorted. Default is ascending.
*    (default = A(scending)) (OPTIONAL)
*
*
************************************************************************
** DATE      INITIALS    MODIFICATION DESCRIPTION
************************************************************************
**
************************************************************************
*  © 2011 Pharmaceutical Product Development, Inc.
*  All Rights Reserved.
************************************************************************************
**********************************;

%global &sysmacroname._RC;


%let currentmacro = &sysmacroname;

%*-----------------------------------------------------------------------------
```

```
--*;
  %PUT ---------------------------------------------- ;
  %PUT INFO: (&SYSMACRONAME) ;
  %PUT INFO: Version 1.0 ;
  %PUT -START----------------------------------------- ;
  %let &SYSMACRONAME._RC =0;
%*-------------------------------------------------------------------------------
--*;

%let abort = no;

%* check for invalid IN_DATA or invalid variables in IN_DATA;
%mu_check_req_parameters
    ( parameters_to_check   =  in_data
    , help                  = no
    ) ;

%** extract byvars and store in byvar1-byvar&num_byvars **;
%if "&byvars" ne "" %then %do;
      %mu_wordscan(string=&byvars, root=byvar, numw=num_byvars);

      %if "&byvars_preloadfmt"="" %then %do;
          %do __idx__=1 %to &num_byvars;
             %let byvars_preloadfmt = &byvars_preloadfmt BLANK;
          %end;
      %end;

%end;

%if "&byvars" ne "" and "&byvars_preloadfmt" ne "" %then %do;
        %let byvars_preloadfmt = %upcase(&byvars_preloadfmt);

        %**extract preload format**;
        %mu_wordscan(string=&byvars_preloadfmt, root=fmt, numw=num_fmts);
        %put num_fmts = &num_fmts;

        %**number of value in &BYVARS_PRELOADFMT should be the same as number of
value in &BYVARS**;
        %if &num_fmts ne &num_byvars %then %do;
                %let &sysmacroname._RC = 1;
                %put ALERT_P: &SYSMACRONAME - The number of values found in
parameter BYVARS and ;
                %put ALERT_P: &SYSMACRONAME - parameter BYVARS_PRELOADFMT do not
match.;
                %put ALERT_P: &SYSMACRONAME - Program will ABORT.;
                %let abort = yes;
                %goto exit;
        %end;
        %do i=1 %to &num_fmts;
            data _null_;
```

```
                call symput("fmt&i", compress("&&fmt&i", '.'));
             run;
        %end;

        %do i=1 %to &num_fmts;
          %**check if fmt&i can be loaded or not, if not then abort the program*;
          proc format lib=work cntlout=tmpfmt
(where=(fmtname=upcase(compress("&&fmt&i",'$')))) ;
          run ;

          %mu_nobs(tmpfmt);

          %if &tmpfmt_nobs = 0 and "&&fmt&i" ne "BLANK" %then %do;
           %let &SYSMACRONAME._RC = 2 ;
           %put ALERT_P: &SYSMACRONAME: Parameter BYVARS_PRELOADFMT contains format
&&fmt&i,;
           %put ALERT_P: &SYSMACRONAME: however the format &&fmt&i cannot be
found/loaded;
           %put ALERT_P: &SYSMACRONAME - Program will ABORT. ;
           %let abort = yes ;
           %goto exit ;
         %end;

       %end;

%end;
%else %if "&byvars" ne "" and "&byvars_preloadfmt" eq "" %then %do;
        %do i=1 %to &num_byvars;
                %global fmt&i;
                %let fmt&i = BLANK;
        %end;
%end;


%*sorting order;
%if "&byvars" ne "" and "&byvars_sort_order" ne "" %then %do;

        %**extract value from byvars_sort_order**;

        %mu_wordscan(string=&byvars_sort_order, root=sort, numw=num_sorts);
        %put num_sorts = &num_sorts;

        %**if number of values in &byvars_sort_order is 1 then assume all variables
will sort with that value**;
        %**so it is allowed to have different number of values in
&byvars_sort_order and &byvars;
        %**otherwise, number should match.;

        %if &num_sorts ne &num_byvars and &num_sorts ne 1 %then %do;
                %let &sysmacroname._rc = 3;
```

```
                %put ALERT_P: &SYSMACRONAME - The number of values found in
parameter BYVARS and ;
                %put ALERT_P: &SYSMACRONAME - parameter BYVARS_SORT_ORDER do not
match.;
                %put ALERT_P: &SYSMACRONAME - Program will ABORT.;
                %let abort = yes;
                %goto exit;
        %end;
        %else %if &num_sorts ne &num_byvars and &num_sorts eq 1 %then %do;
                %put ALERT_I: &SYSMACRONAME - The number of values found in
parameter BYVARS (&num_byvars) and ;
                %put ALERT_I: &SYSMACRONAME - parameter BYVARS_SORT_ORDER contains
1 value (&byvars_sort_order);
                %put ALERT_I: &SYSMACRONAME - Program will assume all variables
sorted with &byvars_sort_order;
                %do i = 2 %to &num_byvars;
                        %global sort&i;
                        %let sort&i = &sort1;
                %end;
        %end;
%end;
%else %if "&byvars" ne "" and "&byvars_sort_order" eq "" %then %do;
        %put ALERT_I: &SYSMACRONAME - BYVARS_SORT_ORDER is not specified.  Macro
will;
        %put ALERT_I: &SYSMACRONAME - sort all BYVARS by internal value;
        %do i=1 %to &num_byvars;
                %global sort&i;
                %let sort&i = I;
        %end;
%end;


%**if sort is F but preload format is BLANK, then check to see if there is format
attached, if not reset sort to I;
%if "&byvars" ne "" %then %do;
        %do __xi__=1 %to &num_byvars;
           %global __fmt&__xi__;
           %**format used to create fmtd_ variables**;
           %let __fmt&__xi__ = &&fmt&__xi__;

           %mu_var_attributes(datasets=&in_data, variables=&&byvar&__xi__);

            data _null_;
                 call symput("muva_in_data", tranwrd("&in_data", ".", "_"));
            run;

            %if &&__fmt&__xi__ = BLANK and &&&&&muva_in_data._&&byvar&__xi__.._FMT
ne %then %do;
               %let __fmt&__xi__ = &&&&&muva_in_data._&&byvar&__xi__.._FMT;
```

```
               %**remove dot from the format**;
               data _null_;
                   call symput("__fmt&__xi__", compress("&&__fmt&__xi__", '.'));
               run;

            %end;

            %if (%upcase(%substr(&&sort&__xi__, 1, 1)) = F and "&&fmt&__xi__" eq
"BLANK") or
                  (%upcase(%substr(&&sort&__xi__, 1, 1)) = N and "&&fmt&__xi__"
eq "BLANK") %then %do;



                  %if &&&&&muva_in_data._&&byvar&__xi__._FMT  = %then %do;
                    %if  %upcase(%substr(&&sort&__xi__, 1, 1)) = F %then %do;
                       %put ALERT_I: &SYSMACRONAME - User asked that variable
&&byvar&__xi__ be sorted by formatted value in parameter ;
                       %put ALERT_I: &SYSMACRONAME - However, since there is no
format provided,;
                       %put ALERT_I: &SYSMACRONAME - &&byvar&__xi__ cannot be
sorted by the formatted value.;
                       %put ALERT_I: &SYSMACRONAME - &&byvar&__xi__ will be sorted
by internal value.;
                    %end;
                    %else %if %upcase(%substr(&&sort&__xi__, 1, 1)) = N %then %do;
                       %put ALERT_I: &SYSMACRONAME - Value specified in
BYVARS_SORT_ORDER for variable &&byvar&__xi__  is N;
                       %put ALERT_I: &SYSMACRONAME - However, there is no format
provided,;
                       %put ALERT_I: &SYSMACRONAME - &&byvar&__xi__ will be sorted
by internal value.;
                    %end;
                     %let sort&__xi__ = I;
                  %end;

            %end;
         %end; /**do __xi__=1 **/
%end;


%**value of valid sort should be I F and N, reset it to I if invalid value found**;
%if "&byvars" ne "" %then %do;
         %do i=1 %to &num_byvars;
                 %if %upcase(%substr(&&sort&i, 1, 1)) ne I and
                         %upcase(%substr(&&sort&i, 1, 1)) ne F and
                         %upcase(%substr(&&sort&i, 1, 1)) ne N %then %do;
                         %put ALERT_I: &SYSMACRONAME - BYVARS_SORT_ORDER is invalid.
 Macro will;
                         %put ALERT_I: &SYSMACRONAME - sort &&byvar&i in INTERNAL
```

```
order;
                        %let sort&i = I;
                        %put sort&i = &&sort&i;
                %end;
        %end;
%end;


%*sort_asc_or_desc;
%if "&byvars" ne "" and "&byvars_sort_asc_or_desc" ne "" %then %do;
        %mu_wordscan(string=&byvars_sort_asc_or_desc, root=asc_or_desc,
numw=num_asc_or_desc, delim=%str( ));
        %put num_asc_or_desc = &num_asc_or_desc;

        %**if only one value is in &SORT_ASC_OR_DESC, then all &BYVAR will use this
value*;
        %**otherwise number of value in &sort_asc_or_desc and in &byvars should
match**;

        %if &num_asc_or_desc ne &num_byvars and &num_asc_or_desc ne 1 %then %do;
                %let &SYSMACRONAME._RC = 4;
                %put ALERT_P: &SYSMACRONAME - The number of values found in
parameter BYVARS and ;
                %put ALERT_P: &SYSMACRONAME - parameter BYVARS_SORT_ASC_OR_DESC do
not match.;
                %put ALERT_P: &SYSMACRONAME - Program will ABORT.;
                %let abort = yes;
                %goto exit;
        %end;
        %else %if &num_asc_or_desc ne &num_byvars and &num_asc_or_desc eq 1 %then
%do;
                %put ALERT_I: &SYSMACRONAME - The number of values found in
parameter BYVARS (&num_byvars) and ;
                %put ALERT_I: &SYSMACRONAME - parameter BYVARS_SORT_ASC_OR_DESC
contains 1 value (&byvars_sort_asc_or_desc);
                %put ALERT_I: &SYSMACRONAME - Program will assume all variables
sorted with &byvars_sort_asc_or_desc;
                %do i = 2 %to &num_byvars;
                        %global asc_or_desc&i;
                        %let asc_or_desc&i = &asc_or_desc1;
                %end;
        %end;
%end;
%else %if "&byvars" ne "" and "&byvars_sort_asc_or_desc" eq "" %then %do;

        %**if &byvars_sort_asc_or_desc is blank, then set it to A**;

        %put ALERT_I: &SYSMACRONAME - BYVARS_SORT_ASC_OR_DESC is not specified.
Macro will;
        %put ALERT_I: &SYSMACRONAME - sort all BYVARS by ascending value;
```

```
        %do i=1 %to &num_byvars;
                %global asc_or_desc&i;
                %let asc_or_desc&i = A;
        %end;
%end;

%**for invalid BYVARS_SORT_ASC_OR_DESC,set it to A;
%if "&byvars" ne "" %then %do;
        %do i=1 %to &num_byvars;
                %if %upcase(%substr(&&asc_or_desc&i, 1, 1)) ne A and
                        %upcase(%substr(&&asc_or_desc&i, 1, 1)) ne D %then %do;
                        %put ALERT_I: &SYSMACRONAME - BYVARS_SORT_ASC_OR_DESC is
invalid.  Macro will;
                        %put ALERT_I: &SYSMACRONAME - sort &&byvar&i in ASCENDING
order;
                        %let asc_or_desc&i = A;
                        %put asc_or_desc&i = &&asc_or_desc&i;
                %end;
        %end;
%end;

    %*create dataset byvar_format&i to be used in md_add_byvar_order**;
    %if "&byvars" ne "" %then %do;
         %do i=1 %to &num_byvars;
             %if "&&__fmt&i" ne "BLANK" %then %do;
                        proc format lib=work

cntlout=byvar_format&i(where=(fmtname=upcase(compress("&&__fmt&i",'$'))));
                        run;
             %end;
         %end;
    %end;

    %if "&byvars" ne "" /*and "&byvars_preloadfmt" ne ""*/ %then %do;
        %* determine if BYVARS_PRELOADFMT is a mulilabel format or not.  If it is,
         then add mlf option in class statement in PROC MEANS below.;

        %* also determine one possible value of the BYVAR variable for situations
with 0 records.;

                %do i=1 %to &num_byvars;
                    %if "&&__fmt&i" ne "BLANK" %then %do;
                        /*
                        proc format lib=work

cntlout=byvar_format&i(where=(fmtname=upcase(compress("&&fmt&i",'$'))));
                        run;
                        */

                        %global byvar_start&i  ;
```

```
                        data byvar_mlf&i;
                            set byvar_format&i;
                            if _n_=1;
                            call symputx("byvar_start&i",start);

                            if index(hlo,'M') gt 0;
                        run;

                        %**only when preloadfmt is used, then do this step**;

                        %*if "&&fmt&i" ne "BLANK" %then %do;
                           %mu_nobs(byvar_mlf&i);
                        %*end;
                                %* determine if byvars_preloadfmt or attached
format is using NOTSORTED option;

                             %if %upcase(%substr(&&sort&i, 1, 1)) = N %then %do;

                                data byvar_notsorted&i;
                                    set byvar_format&i;
                                    if index(hlo,'S') gt 0;
                                run;

                                        %mu_nobs(byvar_notsorted&i);

                                        %if &&byvar_notsorted&i._nobs = 0 %then
%do;
                                                %put ALERT_I: &SYSMACRONAME:
Parameter BYVAR_SORT_ORDER has N (NOTSORTED) specified. However, ;
                                                %put ALERT_I: &SYSMACRONAME: the
format &&__fmt&i was not created using the NOTSORTED option.;
                                                %put ALERT_I: &SYSMACRONAME:
option. BYVAR_SORT_ORDER will default to I (INTERNAL).;
                                                %let sort&i = I;
                                        %end;
                             %end;
                    %end;
                    %else %do;
                                %if %upcase(%substr(&&sort&i, 1, 1)) = N  %then
%do;
                                                %put ALERT_I: &SYSMACRONAME:
Parameter BYVAR_SORT_ORDER has N (NOTSORTED) specified. However, ;
                                                %put ALERT_I: &SYSMACRONAME: there
is no associated format provided.;
                                                %put ALERT_I: &SYSMACRONAME:
BYVAR_SORT_ORDER will default to I (INTERNAL) for variable &&byvar&i...;
                                                %let sort&i = I;
                                %end;
```

```
                    %end;
                %end;
    %end;


%PUT ------------------------------------------------ ;
%PUT INFO: (&SYSMACRONAME) ;
%PUT INFO: Version 1.0 ;
%PUT END;
%PUT ------------------------------------------------ ;

%exit:
%if &abort = yes %then %do;
    data _null_;
        abort;
    run;
%end;

%mend md_byvar_check;
```