

```

*****
*****A macro that execute R script in base SAS*****
** MACRO Version: 1.0 **
** SAS Version: 9.1.3/9.2 **
** R Version: 2.14.0 **
** Date: Nov 24, 2011 **
** Author: Xin Wei, Ph.D. **
** Affiliation: Roche Pharmaceuticals, INC **
** Instruction: This SAS macro enables native R language to be embedded in and executed along with a SAS program **
** in the Base SAS environment under Windows OS. This macro executes a user-defined R code in batch **
** mode by calling the unnamed pipe method within base SAS. The R textual and graphical output can be **
** routed to the SAS output window and result viewer, respectively. Also, this macro automatically **
** converts data between SAS datasets and R data frames such that the data and results from each **
** statistical environment can be utilized by the other environment. The objective of this macro is **
** to leverage the strength of the R programming language within the SAS environment in a systematic **
** manner. Moreover, this macro helps statistical programmers to learn a new statistical language **
** while staying in a familiar environment. The macro variables are described as follows: **
** SAS2R specifies the names of SAS datasets to be converted to R dataframe. Can be single file name or **
** multiple files whose names are separated by space. **
** R2SAS specifies the names of R dataframes to be converted to SAS datasets. Can be single file name or **
** multiple files whose names are separated by space. **
** rpath The full path and file names of R executable file for various R version from 2.11 to 2.14 **
*****;

```

```

options nomerror nomprint nomlogic nosymbolgen;
options nonotes;
%macro proc_R(SAS2R=, R2SAS=);
options notes;
%GLOBAL _SAS2R _R2SAS saswork r_code;
%let _SAS2R=&SAS2R;
%let _R2SAS=&R2SAS;
%let saswork=%sysfunc(pathname(work));
data _null_;
  call symput('r_code','r_code' || trim(left(scan(put(datetime(),best.),1,'.'))));
run;
%put &r_code;
data _null_;
  file "&saswork\&r_code..r";
  infile cards4;
  input;
  put _infile_;
%mend Proc_R;

%macro run_R(rpath=%str(C:\Progra~1\R\R-3.5.2\bin\R.exe));

run;

```

```

data front_code after_code body_code r_code Rlog file; delete; run;
data body_code;
  infile "&saswork\&r_code..r" length=len;
  input var1 $varying8192. len;
run;
data _null_;
  call symput('sasdirec', "&saswork");
  call symput('rdirec', trim(left(tranwrd("&saswork", '\', '/'))));
  call symput('rsaswork', trim(left(tranwrd("&saswork", '\', '/'))));
run;
*****determine if customer defined code produce fixed figure*****;
%let crfg=0;
%let userdefloc=0;
data _null_;
  set body_code;
  if index(lowercase(var1), 'setwd') and
  (index(lowercase(var1), '#')=0 or index(lowercase(var1), '#')> index(lowercase(var1), 'setwd')) then do;
    call symput('userdefloc', '1');
  directory=scan(var1, 2, '(');
  directory=compress(directory, '()');
  directory=compress(directory, '"');
  call symput('rdirec', trim(left(directory)));
  directory=tranwrd(directory, '/', '\');
  call symput('sasdirec', trim(left(directory)));
  end;
  if index(lowercase(var1), '.gif') or index(lowercase(var1), '.jpeg') or index(lowercase(var1), '.jpg')
  or index(lowercase(var1), '.png') or index(lowercase(var1), '.ps') then call symput('crfg', '1');
run;
*****convert sas file to csv for R import*****;
%let i=1;
%do %while(%scan(&_sas2R, &i, %str( )) ne);
  %let transfer=%scan(&_sas2R, &i, %str( ));
  %let i=%eval(&i+1);
  proc export data=&transfer outfile="&sasdirec/&transfer..csv" replace; run;
%end;

****get current time before run batch R***;
data _null_;
  call symput('nowtime', trim(left(scan(put(datetime(), best.), 1, '.'))));
run;
%put &&&&&nowtime;

data front_code;
  length var1 $1000;
  format var1 $1000.;

```

```

        %if &userdefloc=0 %then %do;
var1='setwd("'||"&rsaswork"||'")'; output;
%end;
var1='library(grDevices)'; output;
%if &crfg=0 %then %do;
    var1='png("'||"&rdirec/_&nowtime"||'.png")'; output;
%end;
*****prepare R code use dataframe to read in SAS dataset via CSV****;
%let i=1;
%do %while(%scan(&_sas2r,&i,%str( )) ne);
%let transfer=%scan(&_sas2r,&i,%str( ));
%let i=%eval(&i+1);
    var1="&transfer"||"<- read.csv("'||"&rdirec/&transfer..csv"||'")";
        output;
%end;
run;

data after_code;
length var1 $1000;
format var1 $1000.;
*****prepare R code to output csv file for sas import****;
%let i=1;
%do %while(%scan(&_R2SAS,&i,%str( )) ne);
%let transfer=%scan(&_R2SAS,&i,%str( ));
%let i=%eval(&i+1);
    var1="write.csv(&transfer,'"||"&transfer..csv"||"',row.names=F)";
        output;
%end;
%if &crfg=0 %then %do;
    var1='dev.off()'; output;
%end;
var1='q()'; output;
run;

*****update R code****;
data r_code;
set front_code body_code after_code;
run;
data _null_;
file "&saswork/&r_code..r";
set r_code;
put var1;
run;

```

```

****get current time before run batch R***;
data _null_;
  call symput('beforetm',trim(left(put(datetime(),datetime19.))));
run;
%put $$$$&beforetm;

%let _saswork=%bquote("&rsaswork/&r_code..r");
%let _rdirec=%bquote("&rdirec/r_log_&nowtime..txt");

options noxwait xsync;
filename proc_r pipe "&rpath CMD BATCH --vanilla --quiet
                    &_saswork &_rdirec";
data _null_;
  infile proc_r;
run;

****get current time after run batch R***;
data _null_;
  call symput('aftertm',trim(left(put(datetime(),datetime19.))));
run;
%put $$$$&aftertm;

data rlog;
  infile "&sasdirec\r_log_&nowtime..txt" length=len;
  input var1 $varying8192. len;
run;
data rlog;
  set rlog;
  var1=trim(left(var1));
run;
title "*****R OUTPUT*****";
proc print data=rlog(rename=(var1=R_OUTPUT_LOG)) noobs; run;
title;

*****display R graphics*****;
%let _sasdirec=%bquote("&sasdirec");
FileName MyDir Pipe "dir &_sasdirec /a:-d";
data file;
  infile MyDir lrecl=300 trunccover;
  input @1 file $100. @;
  format file $100.;
  crtime=substr(file,1,20);
  if trim(left(scan(lowercase(file),2,'.'))) in ('gif','png','jpeg','jpg','ps') then do;
    _crttime=input(crtime,mdyampm.);
  end;
run;

```

```

        temp=tranwrd(file,trim(left(crtime)), '*');
        temp=scan(temp,1, '*');
        filename=trim(left(scan(temp,2, ' ')));
    end;
run;

proc sort data=file; by descending _crtime descending filename;
    where trim(left(scan(lowercase(file),2, '.'))) in ('gif', 'png', 'jpeg', 'ps');
run;
data _null_;
    set file;
        if _n_=1 then do;
            call symput('fgsw',put(input("&beforetm",datetime19.)<(input(crtime,mdyampm.)+60),best.));
            temp=tranwrd(file,trim(left(crtime)), '*');
                temp2=scan(temp,1, '*');
                ffname=scan(temp2,2, ' ');
                call symput('ffname',trim(left(ffname)));
        end;
run;
%put $$$$&ffname;
%if &fgsw=1 %then %do;

/*
ODS ESCAPECHAR='^';
ODS HTML FILE="&sasdirec\rhtml_&nowtime..html" STYLE=minimal
GPATH="&sasdirec\" GTITLE GFOOTNOTE;
ods listing close;
%global inhtml;
%let inhtml=%bquote("&sasdirec\&ffname");
DATA _NULL_;
FILE PRINT;
PUT "<IMG SRC=&inhtml BORDER='0'>";
RUN;
ods html close;
ods listing;
*/

Data _NULL_;
    dcl odsout obj();
    obj.image(file:"&inhtml");
run;

/*
options reset=all device=win;
options iback="&rdirec\&ffname" imagestyle=fit;

```

```
proc gslide;
run;
quit;
*/

%end;
*****convert csv file from R to SAS file*****;
%let i=1;
%do %while(%scan(&_R2SAS,&i,%str( )) ne);
    %let transfer=%scan(&_R2SAS,&i,%str( ));
    %let i=%eval(&i+1);
    proc import datafile="&sasdirec/&transfer..csv" out=&transfer replace; guessingrows=1000; run;
%end;

*options notes;
%mend;
```