

```

*****;
* Project          : ZRHM-REXA-07-JP
*
* Program name     : F1501020501.sas
*
* Author           : L. Yan
*
* Date created      : 05/20/2015
*
* Purpose          : F1501020501
*
* Revision History :
*
* Date            Author      Ref      Revision (Date in YYYYMMDD format)
*
*****;

%let prgname=F1501020501_ZRHM_REXA_07_JP_V1;
options mprint;

options sasautos=("W:\pmp07\macros" sasautos) notes;
%init(delivery=9);

%titlecsv(prgname=&prgname., version=1);

%put &title1;
%put &title2;
%put &APPENDIX;
%put &endpoint;
%put &outname.;
%put &repversion.;

options missing="";

%macro cal_summary_pvalue(wher=, outnum=, method=, used=, var=, in=, pflg=, paramcd=, avisit=);
title2 h=10pt j=1 "&used";

proc sort data=&in. out=anadt_&outnum.;
by usubjid;
where &wher. ;
run;

proc sort data=anadt_&outnum.;
by trtcd;
run;

%if &method = 1 %then %do;
title3 h=10pt j=1 "Paramcd: &paramcd, &avisit. Model: Mixed, Method: Log";

proc means data = anadt_&outnum. noprint;
by trtcd;
var &var.;
output out=xlab_&outnum. n=n mean=mean median=med std=sd min=min max=max q1=q1 q3=q3 lclm=lclm uclm=uclm;
run;
%end;

%if &method = 2 %then %do;
title3 h=10pt j=1 "Paramcd: &paramcd, &avisit. Model: Mixed, Method: Normal";

proc means data = anadt_&outnum. noprint;
by trtcd;
var aval;
output out=xlab_&outnum. n=n mean=mean median=med std=sd min=min max=max q1=q1 q3=q3 lclm=lclm uclm=uclm;
run;
%end;

data xlab_&outnum.;
set xlab_&outnum.;
n1 = trim(left(compress(put(n, 8.))));
if sd > . then mean1 = (trim(left(compress(put(mean, 8.1))))||' ( '|trim(left(compress(put(ceil(sd*100)/100, 8.2))))
)||')';
else mean1 = (trim(left(compress(put(mean, 8.1))))||' (NA)';
ci1=trim(left(compress(put(floor(lclm*100)/100, 8.2))))||', '|trim(left(compress(put(ceil(uclm*100)/100, 8.2)))));
median1 = trim(left(compress(put(med, 8.1)))));
q1q3 = trim(left(compress(put(q1, 8.2))))||', '|trim(left(compress(put(q3, 8.2)))));
min1 = trim(left(compress(put(min, 8.))))||', '|trim(left(compress(put(max, 8.0)))));

run;

/*
proc mixed data=anadt_&outnum.;

Class trtcd sex UCPDGR1;

```

```

Model logaval = logbase sex UCPDGR1 trtcd / outp=pred;

lsmeans trtcd / pdiff =control('mCC') alpha=0.05 cl;

ods output lsmeans=lsmeans_&outnum. (keep=trtcd lower upper estimate); *each arm;

ods output diffs=LSMeanDiffCL&outnum. (keep=trtcd lower upper probt estimate); * lsmean and C.I. for ratios;

ods output covparms=ROOTMSE&outnum.(rename=(estimate=mse)); *MSE;

run;
*/

proc mixed data=anadt_&outnum.;
class trtp sex UCPDGR1;

%if &method = 1 %then %do;
model logaval = logbase sex UCPDGR1 trtp/ outp=pred;
%end;
%if &method = 2 %then %do;
model aval = base sex UCPDGR1 trtp/ outp=pred;
%end;
lsmeans trtp / pdiff =control('mCC') alpha=0.05 cl;
lsmeans trtp / pdiff =control('SA') alpha=0.05 cl;
ods output lsmeans=lsmeans_&outnum. (keep=trtp lower upper estimate); *each arm;
ods output diffs=LSMeanDiffCL&outnum. (keep=_trtp trtp lower upper probt estimate where=(TRTP="THSm2.2")); * lsmean and
C.I. for ratios;
ods output covparms=estimate&outnum.(rename=(estimate=rootmse)); *MSE;
run;
ods output close;

data pval&outnum.;
set LSMeanDiffCL&outnum.;
ProbtDiff=probt;
keep trtp ProbtDiff;
run;

data lsmeans_&outnum.;
set lsmeans_&outnum.;
lowercl=lower;
uppercl=upper;
lsmean=estimate;
keep trtp lowercl uppercl lsmean;
run;

data LSMeanDiffCL&outnum.;
set LSMeanDiffCL&outnum.;
lowercl=lower;
uppercl=upper;
difference=estimate;
keep trtp _trtp lowercl uppercl difference;
run;

data lsmeans_&outnum.;
set lsmeans_&outnum.;
if TRTP="THSm2.2" then trtcd=1;
else if TRTP="mCC" then trtcd=2;
else if TRTP="SA" then trtcd=3;

%if &method = 1 %then %do;
Estimate1 = exp(lsmean); /* Ratio of geometric mean */
LowerCL = exp(lowercl); /* 95% CI lower bound */
UpperCL = exp(uppercl); /* 95% CI upper bound */
%end;
%if &method = 2 %then %do;
Estimate1 = lsmean; /* Ratio of geometric mean */
LowerCL = lowercl; /* 95% CI lower bound */
UpperCL = uppercl; /* 95% CI upper bound */
%end;

run;

data ROOTMSE&outnum.;
set estimate&outnum.;
*CVperc=100*sqrt(exp(rootmse**2)-1);
cvperc=100*sqrt(exp(rootmse)-1);
run;

proc sort data=lsmeans_&outnum. nodupkey;
by trtcd;
run;

```

```

data lsmeans_&outnum.;
length geomean geoci $100;
set lsmeans_&outnum.;
geomean=strip(put(ESTIMATE1, 8.2));
geoci=strip(put(floor(LowerCL*100)/100, 8.2)||", "||strip(put(ceil(UpperCL*100)/100, 8.2)));

keep trtcd geomean geoci;
run;

proc sort data=LSMeanDiffCL&outnum. nodup;
by TRTP _TRTP;
run;

data LSMeanDiffCL&outnum.;
set LSMeanDiffCL&outnum.;
myord=1;
run;

data ROOTMSE&outnum.;
set ROOTMSE&outnum.;
myord=1;
run;

data LSMeanDiffCL&outnum.;
merge LSMeanDiffCL&outnum. ROOTMSE&outnum.;
by myord;
run;

data LSMeanDiffCL&outnum.;
length geomean geoci $100;
set LSMeanDiffCL&outnum.;
if _TRTP eq "mCC" then trtcd=4;
if _TRTP eq "SA" then trtcd=5;

%if &method = 1 %then %do;

    difference = 100*exp(difference); /* Ratio of geometric mean */
    lowercl = 100*exp(lowercl); /* 95% CI lower bound */
    uppercl = 100*exp(uppercl); /* 95% CI upper bound */
%end;

%if &method = 2 %then %do;

    difference =difference; /* Ratio of geometric mean */
    lowercl = lowercl; /* 95% CI lower bound */
    uppercl = uppercl; /* 95% CI upper bound */
%end;

%if &method = 1 %then %do;
geomean=strip(put(round(difference, 0.01), 8.2))||" ("||strip(put(ceil(CVperc*100)/100, 8.2)) ||")";
%end;
%if &method = 2 %then %do;
geomean=strip(put(round(difference, 0.01), 8.2))/" ("||strip(put(CVperc, 8.1)) ||")"/;
%end;

geoci=strip(put(floor(lowercl*100)/100, 8.2))||", "||strip(put(ceil(uppercl*100)/100, 8.2)));
keep trtcd geomean geoci;
run;

data pval&outnum.;
set pval&outnum.;
if _n_=1;
trtcd=4;
keep trtcd ProbtDiff;
format ProbtDiff PVALUE6.3;
run;

proc sort data=LSMeanDiffCL&outnum.;
by trtcd;
run;

data LSMeanDiffCL&outnum.;
merge LSMeanDiffCL&outnum. pval&outnum.;
by trtcd;
run;

data LSMeanDiffCL&outnum.;
set LSMeanDiffCL&outnum.;
%if &method = 1 %then %do;
    if (difference < 100) then ProbtDiff=ProbtDiff/2;
    else ProbtDiff=1-ProbtDiff/2;
%end;
%if &method = 2 %then %do;
/*if (difference < 0) then*/ ProbtDiff=ProbtDiff/2;
* else ProbtDiff=1-ProbtDiff/2;
%end;

```

```

run;

data mrep_&outnum.;
set lsmeans_&outnum. LSMeanDiffCL&outnum.;
run;

proc sort data=mrep_&outnum.;
by trtcd;
run;

proc sort data=xlab_&outnum.;
by trtcd;
run;

data xlab_&outnum.;
merge xlab_&outnum.(in=a) mrep_&outnum.;
by trtcd;
run;

data xlab_&outnum.;
set xlab_&outnum.;
if geoci ne "";
run;

proc transpose data = xlab_&outnum. out=xlab_1_&outnum.;
  id trtcd;
  var n1 geomean geoci ProbtDiff;
run;

data rep_&outnum.;
length _name_ _1 _2 _3 ord1 $100;
set xlab_1_&outnum.;
ord1="&outnum.";
ordnum=input(ord1, best.);
if upcase(_name_)="N1" then do; _name_="n"; sord=0; end;
%if &method = 1 %then %do;
if upcase(_name_)="GEOMEAN" then do; _name_="Geometric LS Mean (CV%)"; sord=1; end;
if upcase(_name_)="GEOCI" then do; _name_="95% CI of Geometric Mean"; sord=2; end;
if upcase(_name_)="CI1" then do; _name_="95% CI of Mean"; sord=4; end;
%end;
%if &method = 2 %then %do;
if upcase(_name_)="GEOMEAN" then do; _name_="LS Mean"; sord=1; end;
if upcase(_name_)="GEOCI" then do; _name_="95% CI"; sord=2; end;
if upcase(_name_)="CI1" then do; _name_="95% CI"; sord=4; end;
%end;

if upcase(_name_)="MEAN1" then do; _name_="Mean (SD)"; sord=3; end;
if upcase(_name_)="MEDIAN1" then do; _name_="Median"; sord=5; end;
if upcase(_name_)="Q1Q3" then do; _name_="Q25, Q75"; sord=6; end;
if upcase(_name_)="MIN1" then do; _name_="Min, Max"; sord=7; end;
if upcase(_name_)="PROBTDIFF" then do; _name_="p-value (one-sided)"; sord=9; end;
run;

data rep;
set rep rep_&outnum.;
run;

%mend;

%macro mainloop(where1=, outn=, where=);

proc sort data=adam.adsl out=trt;
by usbjid;
where PPROT1FL="Y";
run;

data trt;
set trt;
if TRT01A="THSm2.2" then trtcd=1;
else if TRT01A="mCC" then trtcd=2;
else if TRT01A="SA" then trtcd=3;
run;

/*
"ADQSSU.PARAMCD in ("QSUFAC1", "QSU FACT2", "QSU Total");
- AVISITN > 100
- PPROTxFL eq "Y" depending on different analysis periods"

*/

data indata1;
length group $4;
set adam.ADQSSU;
group="A";

```

```

paramn=paramn+100;
used="The where clause used on the dataset adam.adbx: PPROTxFL=Y and ANLO1FL=Y";
if PARAMCD in ("QSUTOTAL") and avisitn >100 and ANLO1FL="Y" and (&where.);
drop DTYPE;
run;

data anldata1;
set indata1;
if paramcd in ("QSUFAC1", "QSUFAC2", "QSUTOTAL") then method=2;
else method=1;
if aval>0 then logaval=log(aval);
if base>0 then logbase=log(base);
run;

proc sort data=anldata1 out=fmt(keep=paramn param) nodupkey;
by paramn param;
run;

proc sort data=anldata1 out=check(keep=method paramcd) nodupkey;
by method paramcd;
run;

data anldata1;
set anldata1;
if &where.;
run;

proc sort data=anldata1 out=check(keep=paramn avisitn avisit method used param paramcd) nodupkey;
by paramn avisitn avisit method;
where &where1.;
run;

data trt_1;
set trt;
run;

data anldata1;
set anldata1;
if TRTP="THSm2.2" then trtcd=1;
else if TRTP="mCC" then trtcd=2;
else if TRTP="SA" then trtcd=3;
run;

data check;
set check;
ord=_n_;
run;

%*cal_summary_pvalue(where=1, outnum=1, var=aval, in=anldata1, pflg=1);

data rep;
run;

data _null_;
set check;
call execute ('%cal_summary_pvalue(where=%str(avisitn=||avisitn|| and paramn=||paramn|| ), outnum=||ord||, met
hod=||method||, used=||used||, var=logaval, in=anldata1, paramcd=||paramcd||, avisit=||avisit||)');
run;

data frep;
set rep;
ord=ORDNUM;
run;

data frep;
merge frep(in=a) check;
by ord;
if a;
if avisitn>.;
run;

proc sort data=trt_1 nodupkey;
by trtcd usubjid;
run;

proc freq data = trt_1 noprint;
tables trtcd/ out= denom;
run;

data _null_;
set denom end=eof;

retain total 0;

total = total+count;

```

```

    if trtcd= 1 then do;
        call symput('trt1', trim(left(put(count,8)))));
    end;
    if trtcd= 2 then do;
        call symput('trt2', trim(left(put(count,8)))));
    end;
    if trtcd= 3 then do;
        call symput('trt3', trim(left(put(count,8)))));
    end;
run;

%put trt1=&trt1 trt2=&trt2 trt3=&trt3;

%macro cal_part_main();

data frep;
set frep;

avisit=propcase(avisit);
if AVISIT="Day 0" then avisit="Baseline";

    %do i = 1 %to 100;
        if (&i-1)*3<ordnum<=&i*3 then pagen=&i;
    %end;

run;

%mend;

%cal_part_main();
data frep&outn.;
set frep;
space=" ";
if _name_="p-value (one-sided)" then delete;
run;

%mend;

%mainloop(where1=%str(method=2), outn=2, where=%str((avisitn<=106 and APUPER=1 and PPROT1FL="Y") or
(avisitn=130 and APUPER=2 and PPROT2FL="Y") or (avisitn=160 and APUPER=3 and PPROT3FL="Y") or (avisitn in (190, 191) /*a
nd APUPER=4*/ and PPROT4FL="Y") ));

data in1;
set frep2;
if _name_="LS Mean";
keep avisitn _4 _5;
run;

data in2;
set frep2;
if _name_="95% CI";
keep avisitn _4 _5;
run;

proc transpose data=in1 out=in1_;
by avisitn;
var _4 _5;
run;

data in1_;
set in1_;
Estimate1=input(col1, best.);
if _name_="_4" then trtcd=1;
if _name_="_5" then trtcd=2;
keep avisitn trtcd Estimate1;
run;

proc transpose data=in2 out=in2_;
by avisitn;
var _4 _5;
run;

data in2_;
set in2_;
LowerCL=input(scan(col1, 1, ", "), best.);
UpperCL=input(scan(col1, 2, ", "), best.);

if _name_="_4" then trtcd=1;
if _name_="_5" then trtcd=2;
keep avisitn trtcd LowerCL UpperCL;
run;

```

```

proc sort data=in1_;
by avisitn trtcd;
run;

proc sort data=in2_;
by avisitn trtcd;
run;

data xlab1;
merge in1_ in2_;
by avisitn trtcd;
run;

%macro mainloop(paramn=, outn=, size=, ypos=95, where=1, method=1);

/* Reshape the data to contain three Y values for */
/* each X for use with the HILOC interpolation. */
data reshape_&paramn._&outn(keep=xvar yvar mean trtcd avisitn Estimate1 LowerCL UpperCL paramn method);
length method 8.;
set xlab1;
paramn=1;
mean=Estimate1;
method="&method";
if avisitn<=106 then xvar=avisitn-100;
else if avisitn=130 then xvar=10;
else if avisitn=160 then xvar=15;
else if avisitn=190 then xvar=20;

yvar=Estimate1;
output;

yvar=LowerCL;
output;

yvar=UpperCL;
output;
run;

proc format;
value visitf
-1=" "
0=" "
6=" "
7=" "
8=" "
11=" "
12=" "
13=" "
14=" "
16=" "
17=" "
18=" "
19=" "
10="30"
15="60"
20="90"
9=" ";
run;

%let annopos=6;
%let xpos=60;
%let yypos=9.5;

data anno3;
length function color $ 8;
retain xsys hsys '3';
ysys="3"; size=0.3;
color="blue"; function='move'; x=30; y=&annopos.; line=1; output;
color="blue"; function='draw'; x=35; y=&annopos.; line=1; output;
color="red"; function='move'; x=60; y=&annopos.; line=3; output;
color="red"; function='draw'; x=65; y=&annopos.; line=3; output;

color="black"; function='move'; x=28; y=&annopos.-2.5; line=1; output;
color="black"; function='draw'; x=87; y=&annopos.-2.5; line=1; output;
color="black"; function='draw'; x=87; y=&annopos.+2; line=1; output;
color="black"; function='draw'; x=28; y=&annopos.+2; line=1; output;
color="black"; function='draw'; x=28; y=&annopos.-2.5; line=1; output;
run;

data anno4;
length function color $8 text style $80;
retain xsys hsys '3';
ysys="3"; position="6";

```

```

color="black"; function='label'; x=2.2; y=27.9-&yypos.+0.3; text="Baseline"; output;
function='label'; x=36; y=&annopos+0.5; color="black"; text="THSm2.2-mCC"; output;
function='label'; x=66; y=&annopos+0.5; color="black"; text="THSm2.2-SA"; output;
size=2.0; color="blue"; function='SYMBOL'; x=32.5; y=&annopos.; style='marker'; text='Z'; output;
size=2.5; color="red"; function='SYMBOL'; x=62.5; y=&annopos.; style='marker'; text='U'; output;

run;

data anno8;
length function color $ 8;
retain xsys '2';
ysys="3"; hsys="3"; size=0.3;
color="black"; function='move'; x=0; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=0; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=1; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=1; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=2; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=2; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=3; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=3; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=4; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=4; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=5; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=5; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=10; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=10; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=15; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=15; y=29+1.05-&yypos.; line=1; output;
color="black"; function='move'; x=20; y=28.3+1.2-&yypos.; line=1; output;
color="black"; function='draw'; x=20; y=29+1.05-&yypos.; line=1; output;

run;

data anno5;
set xlab1;
length function color $8 text style $80 size 8;
retain xsys '2';
ysys="2"; hsys="3"; position="6";
y=Estimate1;
if avisitn<=106 then x=avisitn-100;
else if avisitn=130 then x=10;
else if avisitn=160 then x=15;
else if avisitn=190 then x=20;
size=2.0;
if trtcd=1 then do; color="blue"; function='SYMBOL'; style='marker'; text='Z'; output; end;
size=2.0;
if trtcd=2 then do; color="red"; function='SYMBOL'; style='marker'; text='U'; output; end;
size=2.0;
if trtcd=3 then do; color="green"; function='SYMBOL'; style='marker'; text='C'; output; end;

run;

data anno_&paramn._&outn;
set anno3 anno4 anno5 anno8;
run;

%mend;

%mainloop(paramn=1, outn=1, size=1000, ypos=95, where=1, method=1);

data ototal;
run;

ods tagsets.ExcelXP path="&csvdata." file="&outname..xml" style=Printer;

%macro excelloop(paramn=, pagen=);

ods tagsets.ExcelXP options(sheet_name="Page &pagen.");

proc print data=Reshape_&paramn._1;
var trtcd avisitn paramn estimate1 lowercl uppercl xvar yvar;
run;
quit;

data ototal;
set ototal Reshape_&paramn._1;
run;

%mend;

%excelloop(paramn=1, pagen= 1);

ods tagsets.ExcelXP close;

```



```

data odata.&prgname;
set reshape_1_1;
if paramn>.;
run;

proc format;
value yfmt
0.1=" "
1=" "
10=" "
100=" "
1000=" "
10000=" "
100000=" ";
run;

%macro call(paramn=, method=, outputn=, hsize=, vsize=, innum=, innum1=, outn=, xlabel=, xmin=, xmax=, ymin=, ymax=, b
y=);

options /*leftmargin=0.2cm topmargin=0.1cm rightmargin=0.2cm bottommargin=1cm*/
nodate nonumber nobyline;
ods results off;

%let startobs = 1;
%let eof = 0;
%let imageCnt = 1;

/*****

* handle graph size -> transform to cm when inches are specified

*****/;

%let vunit=%upcase(%scan(&vsize,-1,' 0123456789. '));
%let hunit=%upcase(%scan(&hsize,-1,' 0123456789. '));
%if &vunit=IN or &vunit=INCH %then %let vsize=%sysevalf(%scan(&vsize,1,%str( icIC))*2.58);
%if &hunit=IN or &hunit=INCH %then %let hsize=%sysevalf(%scan(&hsize,1,%str( icIC))*2.58);
%if &vunit=CM %then %let vsize=%sysevalf(%scan(&vsize,1,%str( icIC))*1);
%if &hunit=CM %then %let hsize=%sysevalf(%scan(&hsize,1,%str( icIC))*1);

ods listing gpath="&opath";
ods graphics on / imageName = "lineplot"
imagefmt = png
border = off
scale = no
reset = index
width = 6 cm
height = 4 cm;
ods escapechar="é";

filename graphout "&opath\&outname._&outputn..png";
goptions reset=all device=png gsfname=graphout ftext="Arial/bold" htext=2.5 hsize=6.25 in vsize=4.9 in;

axis1 offset=(2 pct,2 pct) label=("Visit Day")
width=1 minor=none major=none origin=(, 1.0in) order=0 to &xmax. by 1
;
%if &method. =1 %then %do;
axis2 /*length=4.8 in */ label=(angle=90 "QSU-brief Total Score Difference")
width=1 minor=none /*order=&ymin. to &ymax. by &by.*/
;

axis3 /*length=4.8 in */
width=1 minor=none /*order=&ymin. to &ymax. by &by.*/ major=none
;
%end;

%if &method. =2 %then %do;
axis2 /*length=4.8 in */ label=(angle=90 "QSU-brief Total Score Difference")
width=1 minor=none order=&ymin. to &ymax. by &by.
;

axis3 /*length=4.8 in */
width=1 minor=none /*order=&ymin. to &ymax. by &by.*/ major=none
;
%end;

/* Define the symbol characteristics */
symbol1 interpol=hiplotj color=blue line=1 w=1 ;
symbol2 interpol=hiplotj color=red line=3 w=1 ;
symbol3 interpol=hiplotj color=green line=33 w=1 ;

/* symbol4 interpol=none color=blue w=4 font=marker value=W;
symbol5 interpol=none color=red w=4 font=marker value=P;
symbol6 interpol=none color=green w=4 font=marker value=D;

```

```

*/

/* Plot the error bars using the HILOCTJ interpolation */
/* and overlay symbols at the means. */

proc gplot data=reshape_&paramn._&outn;
    plot
        yvar*xvar=trtcd
        /haxis=axis1 vaxis=axis2 anno=anno_&paramn._&outn nolegend;
    format xvar visitf.;
    /* plot
        Estimate1*xvar=trtcd
        /haxis=axis1 vaxis=axis2 nolegend;
    format xvar visitf.;
*/
run;
quit;

proc greplay igout=work.gseg nofs;
delete _all_;
run;
quit;

ods listing close;
ods graphics off;
%mend;

%cal1(paramn=1, method=2, outputn= 1, hsize=4 in, vsize=6 in, innum=1, innum1=5, outn=1, xmin=0, xmax=20, ymin=-3, yma
x=2, by= 1);

%macro rtfoutput;
option nobyline nodate nonumber orientation=landscape;

ods listing close;
%trtrtf_fig(pgmname=&outname., pgmid=1, new=0, style=, bookmark=%lowercase(&outname.));
ods escapechar="é";

title;

%let n_plots=1;
%let orient=landscape;

data _rmtext;
    format text $12.;
    text = 'éR\par\ ''';
    %do i = 1 %to &n_plots;
        output;
    %end;
run;

data _rmtext;
set _rmtext;
pagen=_n_;
run;

%local tblwidth;
%let tblwidth = 6.25;

%do i = 1 %to &n_plots;

proc report data = _rmtext nowd;
    column pagen text;
    where pagen = &i;
    define pagen /order order=internal noprint;
    define text / display style(column)=[cellwidth=&tblwidth.in] ' ';
    compute text;
        *line_count + 1;
        * if line_count = &i then do;
            call define(_row_, 'STYLE', %str(%')STYLE=[just=center postimage="&opath\&&outname._&i..png"%str(%')]);
        * end;
    endcomp;

compute before pagen /style=[fontweight=bold fontsize=3.75];
line @1 "&title1 &title2";
endcomp;

compute after pagen /style=[fontsize=1.75];
line @1 "Note: mCC = Conventional menthol cigarettes; SA = Smoking abstinence; THSm2.2 = Tobacco Heating System 2.2 Ment
hol.";
line @1 "Note: Baseline is the last assessment prior to first product use in CC/THS 2.2 arms on Day 1 or last assessment
prior to 06:29 AM in SA arm on Day 1.";
line @1 "Note: QSU-brief scores reported on a 7-point scale. Higher values indicate greater intensity of urge.";
line @1 " ";

```

```
line @1 "&APPENDIX.";
line @1 "Study ID: ZRHM-REXA-07-JP";
line @1 "Program: &fprgname..sas      Status: &repversion./&fdate.      Page &i. of 1";
endcomp;

run;

%end;

/*
proc datasets nolist;
    delete _rmttext;
quit;
*/
ods rtf close;
ods listing;

%mend;
%rtfoutput;
```