

/\*-----

Covance Study ID : COV-000000106343

Program Name : t\_cyp2a6\_excl\_fas.sas

Purpose : Table 15.2.4.62.2.1(Descriptive Statistics of CYP2A6 ACTIVITY (%) - FAS;

Author : cvn\_pshe

Date of Creation : 11MAY015

Input Data : ADAM.ADSL, ADAM.ADBX

Output Data :

Macros Called :

-----

#### Modification History

-----

Modified by :

Modification Date :

Modification Description:

-----\*/

proc datasets lib=work kill memtype=data nolist;

run;

%m\_printto;

options notes nosource;

```
options notes source source2 nofullstimer validvarname=upcase missing=' ';
```

```
ods _all_ close;
```

```
ods listing;
```

```
*=====;
```

```
* START OF PROGRAM CODE                               ;
```

```
*=====;
```

```
%let tflno=T_15_02_04_62_02_01;
```

```
%let TFL_Part=%scan(&_amp;_SASPROGRAMFILE,-3,%str(/));
```

```
data _null_;
```

```
    tmp("&TFL_Part";
```

```
        if tmp not in ("dev" "qc") then call symput("TFL_Part", "prod");
```

```
        call symput('TFLpath', compress("&_SASPROGRAMFILE", ""));
```

```
            call  
symput('TFLprg',reverse(scan(strip(reverse(compress("&_SASPROGRAMFILE", ""))),1,"/")));
```

```
run;
```

```
*****,
```

```
* read in data ;
```

```
*****,
```

```
/*Use ADSL to get N values for column headers*/
```

```
data adsl;
```

```

        set adam.adsl(where=(fasfl='Y'));

run;

proc sort data=adsl nodupkey out=adsl1;

        by trt01an trt01a subjid;

run;

proc freq data=adsl1(where=(not missing(trt01an))) noprint;

        table trt01an*trt01a/ out =tot(drop=percent rename=(count=total));

run;

data tot2;

        set tot;

        call symput('trt' || compress(put(trt01an,best.)), compress(put(total, best.)));

run;

/*Bring in appropriate data from ADBX*/

data adbx1;

        set adam.adbx(where=(anl02fl='Y' and anl03fl='Y' and fasfl='Y' and paramcd in ('CYP2A6')));

run;

data adbx;

        set adbx1;

        if ablfl='Y' then avisit='Baseline';

run;

```

```
data adbx_orig;  
    set adbx;  
    statval=aval;  
  
    type='abs';  
  
    output;  
  
    statval=pchg;  
  
    type='pchg';  
  
    output;  
run;
```

```
proc sort data=adbx_orig;  
    by type trtan trta avisitn avisit atptn atpt;  
run;
```

```
proc means data=adbx_orig noprint;  
    var statval;  
  
    by type trtan trta avisitn avisit atptn atpt;  
  
    output out=results02 n=n1 mean=mean1 std=sd1 median=median1 min=min1 max=max1 q1=q1  
    q3=q3 lclm=lci1 uclm=uci1;  
run;
```

```
data results03;  
    set results02;  
  
    attrib meansd minmax n median quart aci missc length=$20.;
```

```

n = left(compress(put(n1,8.)));

*for <missing, n(%>;

                                if trtan=3 then do;

                                                if &trt3.=n1 then

missc="";

                                                else

missc=strip(put((&trt3.- n1), 8.)) || ' (' || strip(put(((&trt3.-n1)*100)/&trt3., 8.1)) || ")";

                                                end;

                                else if trtan=4 then do;

                                                if &trt4.=n1 then

missc="";

                                                else

missc=strip(put((&trt4.- n1), 8.)) || ' (' || strip(put(((&trt4.-n1)*100)/&trt4., 8.1)) || ")";

                                                end;

                                else if trtan=5 then do;

                                                if &trt5.=n1

then missc="";

                                                else

missc=strip(put((&trt5.-n1), 8.)) || ' (' || strip(put(((&trt5.-n1)*100)/&trt5., 8.1)) || ")";

                                                end;

IF NOT MISSING(MEDIAN1) THEN MEDIAN = LEFT(COMPRESS(PUT(ROUND(MEDIAN1,0.001),10.3)));

IF NOT MISSING(MEAN1) AND NOT MISSING(SD1) THEN meansd =
LEFT(COMPRESS(PUT(ROUND(MEAN1,0.001),10.3))) || "
(" || STRIP(PUT(0.0001*CEIL(SD1/0.0001),10.4)) || ")";

IF NOT MISSING(MIN1) AND NOT MISSING(MAX1) THEN minmax = strip(put(min1, 10.2)) || ",
" || strip(put(max1, 10.2));

```

```
IF NOT MISSING(Q1) AND NOT MISSING(Q3) THEN QUART =  
LEFT(COMPRESS(PUT(ROUND(Q1,0.001),10.3))) || ' ' ||  
LEFT(COMPRESS(PUT(ROUND(Q3,0.001),10.3))));
```

```
IF NOT MISSING(LCI1) AND NOT MISSING(UCI1) THEN ACI =  
STRIP(PUT(0.001*FLOOR(LCI1/0.001),10.3)) || ' ' || STRIP(PUT(0.001*CEIL(UCI1/0.001),10.3));
```

```
drop /*n1*/ mean1 sd1 median1 min1 max1 q1 q3 uci1 lci1 ;  
  
run;
```

```
/*Obtain subjects with values BLOQ*/
```

```
data adbx_blq;  
  
    set adbx;  
  
    where AQLFL='Y';  
  
    statval=aval;  
  
type='abs';  
  
output;  
  
statsval=pchg;  
  
type='pch';  
  
output;  
  
run;
```

```
proc freq data=adbx_blq noprint;  
  
    table type*trtan*trta*avisitn*avisit*atptn*atpt/ out =blq(drop=percent);  
  
run;
```

```
%macro outrtf(blankn=, halfblnk=);
```

```
%if &halfblnk=N %then %let halfblnk=;
```

```
%else %if &halfblnk=Y %then %let halfblnk=~;
```

```
%let dsid=%sysfunc(open(blq));
```

```
%let nsum=%sysfunc(attrn(&dsid.,nobs));
```

```
%let rc=%sysfunc(close(&dsid.));
```

```
%put "Check " &nsum.;
```

```
%if &nsum. lt 1 %then %do;
```

```
proc sort data=adbx_orig nodupkey out=tpts(keep=type avisitn avisit atptn atpt trtan  
trta);
```

```
by trtan trta type avisitn avisit atptn atpt;
```

```
run;
```

```
data blq1;
```

```
set tpts;
```

```
attrib blq length=$50.;
```

```
IF TYPE='abs' THEN blq='0';
```

```
run;
```

```
%end;
```

```
%else %do;
```

```
PROC SORT DATA=ADBX_ORIG NODUPKEY OUT=TPTS(KEEP=TYPE PARAMN PARAM AVISITN AVISIT  
ATPTN ATPT TRTAN TRTA);
```

```
    BY TRTAN TRTA TYPE PARAMN PARAM AVISITN AVISIT ATPTN ATPT;
```

```
RUN;
```

```
DATA BLQTOTS;
```

```
    SET RESULTS03(RENAME=(N1=TOTAL));
```

```
    KEEP TYPE TR: AVISIT: ATPT: TOTAL;
```

```
RUN;
```

```
PROC SORT DATA=BLQTOTS; by TYPE trtan trta AVISITN AVISIT; RUN;
```

```
PROC SORT DATA=TPTS; by TYPE trtan trta PARAMN PARAM AVISITN AVISIT; RUN;
```

```
DATA TOT_BLQ;
```

```
    MERGE TPTS (drop=atpt atptn) BLQTOTS;
```

```
    BY TYPE TRTAN TRTA AVISITN AVISIT;
```

```
RUN;
```

```
PROC SORT DATA=TOT_BLQ;
```

```
    BY TYPE PARAMN TRTA AVISITN AVISIT ATPTN ATPT;
```

```
RUN;
```

```
DATA BLQ1;
```

```
    ATTRIB BLQ LENGTH=$50.;
```

```
    MERGE BLQ(IN=A) TOT_BLQ;
```

```
    BY TYPE TRTAN TRTA AVISITN AVISIT ATPTN ATPT;
```



```
IF NOT A THEN DO;
```

```
    COUNT=0;
```

```
END;
```

```
if trtan=3 and not missing (count) then percent=count*100/total;
```

```
IF COUNT=0 THEN BLQ='0';
```

```
ELSE IF PERCENT=100 THEN BLQ= PUT(COUNT,3.) || ' (100%)';
```

```
ELSE BLQ=PUT(COUNT,3.) || '  
( ' || LEFT(STRIP(PUT(ROUND(PERCENT,0.1),5.1))) || ' )';
```

```
IF TYPE='pch' THEN BLQ='';
```

```
RUN;
```

```
%end;
```

```
/*Obtain the geometric mean*/
```

```
data gmean;
```

```
    set adbx_orig(where=(type='abs'));
```

```
    statval1=statval;
```

```
    if aval > 0 then ln_statval1=log(statval1);
```

```
run;
```

```
proc means data=gmean noprint;
```

```
    output out=gmean1 mean=mean std=std1 lclm=lci1 uclm=uci1 nmiss=miss;
```

```
    var ln_statval1;
```

```

by trtan trta type avisitn avisit atptn atpt;

run;

data gmean2;

set gmean1;

gmean1=exp(mean);

if miss=0 then do;

    gmean=left(compress(put(gmean1,8.3)));

    if not missing(std1) then gcv=compress(put(0.0001*ceil((sqrt(exp(std1*std1)-
1)*100)/0.0001),8.4));

    if not missing(lci1) then glci=exp(lci1);

    if not missing(uci1) then guci=exp(uci1);

end;

keep type trtan trta avisitn avisit atptn atpt gmean gcv glci guci std1 miss;

run;

```

```

/*Combine the Gmean and BLQ with other stats*/

```

```

proc sort data=results03;

by trtan trta type avisitn avisit atptn atpt;

run;

```

```

proc sort data=blq1;

by trtan trta type avisitn avisit atptn atpt;

run;

```

```

data results04;

merge results03 gmean2 blq1;

    attrib gmeancv length=$20.;

by trtan trta type avisitn avisit atptn atpt;

IF MISS=0 THEN DO;

    if not missing(gcv) then gmeancv=left(trim(gmean)) || ' (' || left(trim(gcv)) || ')';

    else gmeancv=left(trim(gmean));

    if not missing(glci) and not missing(guci) then ci = strip(strip(put(0.001*floor(glci/0.001),8.3)) ||
', ' || strip(put(0.001*ceil(guci/0.001),8.3)));

END;

run;


proc sort data=results04;

    by trtan trta type avisitn avisit atptn atpt;

run;


proc sort data=blq1;

    by trtan trta type avisitn avisit atptn atpt;

run;


data results05;

merge results04 (drop=blq count param paramn total percent) blq1;

    by trtan trta type avisitn avisit atptn atpt;

run;


proc sort data=results05;

```

```
by type avisitn avisit atptn atpt;
```

```
run;
```

```
proc transpose data=results05(where=(type='abs')) out=results06 prefix=r name=varname;
```

```
by avisitn avisit atptn atpt;
```

```
var n meansd median minmax aci quart blq gmeancv missc ci;
```

```
id trtan;
```

```
idlabel trta;
```

```
run;
```

```
proc transpose data=results05(where=(type='pch' and avisitn>100)) out=results06c prefix=c  
name=varname;
```

```
by avisitn avisit atptn atpt;
```

```
var n meansd median minmax aci quart missc blq;
```

```
id trtan;
```

```
idlabel trta;
```

```
run;
```

```
proc sort data=results06;
```

```
by avisitn avisit atptn atpt varname;
```

```
run;
```

```
proc sort data=results06c;
```

```
by avisitn avisit atptn atpt varname;
```

```
run;
```

```

data results07;

merge results06 results06c;

by avisitn avisit atptn atpt varname;

attrib stat variable length = $100.;

varname=upcase(varname);


variable=propcase(avisit);


if varname='N' then do; statord=1; stat='n'; end;

            if varname='MISSC' then do; statord=2; stat='Missing, n (%)'; end;

        else if varname='BLQ' then do;

statord=2.2;

stat='BLOQ, n (%)';

/*                      delete;*/

        end;

        else if varname='GMEANCV' then do;

statord=3;

stat='Geometric Mean (CV%)';

        end;

else if varname='CI' then do;

statord=4;

stat='95% CI of Geometric Mean';

end;

else if varname='MEDIAN' then do;

statord=5;

```

```
    stat='Median';  
end;  
else if varname='QUART' then do;  
    statord=6;  
    stat='Q25, Q75';  
end;  
else if varname='MINMAX' then do;  
    statord=7;  
    stat='Min, Max';  
end;  
else if varname='MEANSD' then do;  
    statord=8;  
    stat='Mean (SD)';  
end;  
else if varname='ACI' then do;  
    statord=9;  
    stat='95% CI of Mean';  
end;  
drop varname;  
run;  
  
data results08;  
set results07;  
if stat='BLOQ, n (%)' and r3='0' and r4='0' and r5='0' then delete;  
run;
```

```
data labels;
```

```
set results08;
```

```
attrib r3 label = "Value"
```

```
    r4 label = "Value"
```

```
    r5 label = "Value"
```

```
    c3 label = '%Change(*)'
```

```
    c4 label = '%Change(*)'
```

```
    c5 label = '%Change(*)';
```

```
    if stat="Missing, n(%)" and r3="" and r4="" and r5="" and c3="" and c4="" and c5="" then delete;
```

```
flag=1;
```

```
if avisitn in (100 106) then period=1;
```

```
    else if avisitn= 190 then period=4;
```

```
run;
```

```
proc sql noprint;
```

```
    create table tflds.&tflno as
```

```
    select period, avisitn, atpt, variable as avisit, statord, stat, r4 as thsm2_2, c4 as thsm2_2_chg, r5  
as mCC, c5 as mCC_chg, r3 as SA, c3 as SA_chg
```

```
    from labels
```

```
    order by avisitn, atptn, statord;
```

```
quit;
```

```
proc sort data=labels;
```

```
    by avisitn atptn statord;
```

```
run;
```

```
data paging;
```

```
set labels;
```

```
by avisitn atptn statord;
```

```
if avisitn=100 then page=1;
```

```
else if avisitn=106 then page=2;
```

```
else if avisitn=190 then page=3;
```

```
call symput("page",compress(put(page,best.)));
```

```
run;
```

```
options number nodate orientation=landscape papersize=Letter missing=' ';
```

```
ods escapechar='$';
```

```
%let linetop = \brdrt\brdrs\brdrw30; * needs to be 1.5pt so calculated in twips (1/20 pt) ;
```

```
%let linebot = \brdrb\brdrs\brdrw30;
```

```
ods path stdlib.t106343 (read) ;
```

```
ods results off;
```

```
ods rtf toc_data file="/cvn/projects/prj/data/000000106343/TFL/&TFL_Part./Tables/&tflno..rtf"  
style=t106343 startpage=yes headery=1440 footery=1440 ;
```

```
ods noproctitle;
```

```
%do i=1 %to &page;
```



```
title ;
```

```
footnote;
```

```
%let wd=0;
```

```
%let subpage=1/*2*/;
```

```
%do j=1 %to &subpage;
```

```
%let maxpage=%eval(&page*&subpage);
```

```
%let npage=%eval(&subpage*&i+&j-&subpage);
```

```
data comp;
```

```
    set paging end=eof;
```

```
        where page=&i;
```

```
/* Amend title as needed */
```

```
_firtitl="Table 15.2.4.62.2.1 Descriptive Statistics of CYP2A6 Activity (%) Excluding Assessments within  
5 Half-Lives of a Concomitant Medication Affecting CYP2A6 Activity - FAS";
```

```
_upcas=(length("Path: &TFLpath.")-  
length(compress("Path:&TFLpath.",'ABCDEFGHIJKLMNOPQRSTUVWXYZ')))/2;
```

```
len=&blankn.-length("(page &npage of &maxpage)");
```

```
if eof then do;
```

```
    call symput('_FSRTITL', trim(left(_firtitl)));
```

```
    call symput('_blankn', compress(put(len,best.)));
```

```

end;

drop _firtitl _upcas len;

run;

ods proclabel = ' ';

ods listing close;

* most set up in template others below;

* title arial 12pt bold with 12pt paragraph space below;

* all headers to be arial 11pt bold;

* data arial 10pt;

* headers to be central, text values left aligned and numeric centered around decimal point;

proc report data = comp missing headline headskip missing nowd split = '$' %if &i=1 and &j=1 %then
%do; contents=' ' %end; %else %do; contents="" %end;;;

        column flag page avisitn atptn variable statord stat

                %if &j=1 %then %do; ("THSm2.2 $(N=&trt4)&linebot" r4 c4) ("mCC$(N=&trt5)&linebot"
r5 c5) ("SA$(N=&trt3)&linebot" r3 c3) %end;;

define flag      / order order = internal noprint;

define page      / order order = internal noprint;

define avisitn   / order order=internal noprint;

define atptn     / order order=internal noprint;

define variable  / group style={just=left cellwidth=3.2cm} style(header)={just=center} "Timepoint";

define statord   / order order = internal noprint;

define stat      / display style={just=left cellwidth=3.2cm} style(header)={just=center} "Statistic";

```

```

%if &j=1 %then %do;

define r3      / display style={just=c cellwidth=2cm} style(header)={just=center};

define c3      / display style={just=c cellwidth=2cm} style(header)={just=center};

define r4      / display style={just=c cellwidth=2cm} style(header)={just=center};

define c4      / display style={just=c cellwidth=2cm} style(header)={just=center};

define r5      / display style={just=c cellwidth=2cm} style(header)={just=center};

define c5      / display style={just=c cellwidth=2cm} style(header)={just=center};

%end;

```

```

break before flag / page %if &i=1 and &j=1 %then %do;

contents="&_fsrtitl" %end; %else %do; contents=" " %end;;

```

```

break after page / page;

```

```

compute after variable;

```

```

    line " ";

```

```

endcomp;

```

```

compute before page / style={protectspecialchars=off};

```

```

    line "&linetop";

```

```

endcomp;

```

```

compute before _page_ / style={just=left protectspecialchars=off};

```

```

    line "\b\fs24\sa24&_FSRTITL." ; * \b = bold, \fs24 is font size 12pt, \sa24 is space after 12pt;

```

```

    line "&linebot";

```

```

endcomp;

compute after _page_/ style={just=left protectspecialchars=off pretext="&linetop."};

    line 'Note: mCC = Menthol conventional cigarettes; SA = Smoking abstinence; THSm2.2
= Tobacco Heating System 2.2 Menthol.';

    line "Note: 'Missing' percentages are based on the number of subjects indicated in the
column header (N), while 'BLOQ' percentages are based on the number of subjects being summarized
(n).";

    line "Note: * % change from baseline, where baseline is defined as the last assessment
prior to first randomized product use in mCC / THS 2.2 Menthol arms or the";

    line "last assessment prior to 10 AM on Day 1 in the SA arm.";

    line ' ';

line "Appendix 15.3.6.20";

    line "Study ID:ZRHM-REXA-08-US Program:&TFLprg Status: &status" &_blankn.*"\~\~"
"&sysdate" &_blankn.*"\~\~" "(Page &i of &page)";

endcomp;

run;

%end;

%end;

ods rtf close;

ods results on;

ods path sashelp.tmplmst (read);

%mend ;

%outrtf(blankn=36, halfblank=N);

```

```
ods listing close;
```

```
proc printto ; run;
```

```
%m_logchk;
```

```
*=====;
```

```
* END OF PROGRAM CODE ;
```

```
*=====;
```