

```

options notes nosource;
proc datasets lib=work nolist memtype=data kill; quit;

/*=====
*Covance Study ID      : 000000106343
*Program Name          : t_oxyst_fas.sas
*Purpose                : Table 15.2.4.66.2 Descriptive Statistics of Oxysterol Parameters  PP set
*Input Data             : adam.adsl, ADAM.adbx
*Output Data            : tflds.T_15_02_04_66_02
*Macros Called          : %m_printto, %mmeans, %outrtf, m_logchk2
*Programmed by          : Ranju Gautam
*Creation Date          : 2015-05-15
*== Modification History ==
*Date      Initials  No. Reason;
*=====*/

%m_printto(ROUTE=yes);

data adsl;
  set adam.adsl(where=(fasfl='Y'));
  if trt01pn=4 then trt=1;
  else if trt01pn=5 then trt=2;
  else if trt01pn=3 then trt=3;
run;

proc freq data=adsl noprint;
  table trt/ out=tot(drop=percent rename=(count=total));
run;

data tot2;
  set tot;
  call symput('trt' || compress(put(trt,best.)), compress(put(total, best.)));
run;

%put THS=&trt1 mCC=&trt2 SA=&trt3;

data adbx1;
  set adam.adbx;
  where anl02f1='Y' AND PARAMCD in ('P6HYDCH' 'P7AHYDCH' 'P56AEPCH' 'P7KETCH' 'P7BHYDCH' 'P56BEPCH' 'P24HYDCH'
    'P25HYDCH' 'P22HYDCH' 'P4BHYDCH' 'P27HYDCH' 'PCHOL')
    AND (fasfl='Y') and avisit in ('Day 0' 'Day 6/Discharge Confinement' 'Day 90');
  if trtpn=4 then trt=1;
  else if trtpn=5 then trt=2;
  else if trtpn=3 then trt=3;
run;

data adbx;
  set adbx1;
  if ablfl='Y' then do;
    avisit='Baseline';
  end;
  if aval not in ( . 0) then logaval=log(aval);
run;

*****;
* macro for general mean stats(n mean std median min max Q25 Q75 lclm uclm) per mock ;
*****;
%macro mmeans(dsn=, class=, var=, out=);

proc means data=&dsn. noprint nway;
  var &var.;
  class &class. trt;
  where AQLFL='Y';
  output out=resultsbn(drop=_type_ _freq_) n=bn;
run;

proc means data=&dsn. noprint nway;
  var &var.;
  class &class. trt;
  output out=resultsbn_(drop=_type_ _freq_) n=n_1;
run;

proc sort data=resultsbn;
  by &class. trt;
run;

```

```

proc sort data=resultsbn_;
  by &class. trt;
run;

data resultsbn1;
  attrib bl length=$20.;
  merge resultsbn(in=a) resultsbn_(in=b);
  by &class. trt;
  if bn ne . and n_1 ne . then p1=put((bn*100)/n_1,11.1);
  if bn ne . then bl=strip(put(bn, best.)) || ' (' || strip(p1)|| " ";
run;

proc means data=&dsn. noprint nway;
  var &var.;
  class &class. trt;
  output out=results02 n=n1 mean=mean1 std=std1 median=median1 min=min1 max=max1 q1=q1 q3=q3 lclm=lci1 uclm=uci1;
run;

data results03;
  set results02;
  attrib meansd length=$30.
    minmax length=$20.
    n length=$20.
    miss length=$20.
    median length=$20.
    quart aci length=$20.;

  n = left(compress(put(n1,8.)));
  if trt=1 then do;
    if &trt1.=n1 then miss="";
    else miss=strip(put((&trt1.-n1), 8.)) || ' (' || strip(put(((&trt1.-n1)*100)/&trt1., 11.1)) || " ";
  end;
  else if trt=2 then do;
    if &trt2.=n1 then miss="";
    else miss=strip(put((&trt2.-n1), 8.)) || ' (' || strip(put(((&trt2.-n1)*100)/&trt2., 11.1)) || " ";
  end;
  else if trt=3 then do;
    if &trt3.=n1 then miss="";
    else miss=strip(put((&trt3.-n1), 8.)) || ' (' || strip(put(((&trt3.-n1)*100)/&trt3., 11.1)) || " ";
  end;

  if not missing(median1) then median = left(compress(put(round(median1,0.01),11.2)));
  if not missing(mean1) and not missing(std1) then meansd = left(compress(put(round(mean1,0.01),11.2)) || ' (' || left(compress(
put(0.001*ceil(std1/0.001),11.3)) || ')');
  if not missing(min1) and not missing(max1) then minmax = left(compress(put(min1,11.1)) || ', ' || left(compress(put(max1,11.1)
)));
  if not missing(lci1) and not missing(uci1) then aci = strip(put(0.01*floor(lci1/0.01),11.2)) || ', ' || strip(put(0.01*ceil(uci1
/0.01),11.2));
  if not missing(q1) and not missing(q3) then quart = strip(strip(put(round(q1, 0.01),11.2)) || ', ' || strip(put(round(q3, 0.01),11.
2))));
  drop n1 std1 median1 min1 max1 q1 q3 ;
run;

data &out._f(keep=param paramn paramcd lci1 uci1 avisitn trt avisit) ;
  set results03;
run;

proc sort data=results03;
  by &class.;
run;

proc sort data=resultsbn1;
  by &class.;
run;

data results04;
  merge results03(in=a) resultsbn1(in=b );
  by &class. trt;
run;

proc sort data=results04;
  by &class. trt;
run;

```

```

proc transpose data=results04(drop=mean1 uci1 lci1 bn) out=&out prefix=r_ name=varname;
  by &class.;
  var n miss bl meansd median minmax aci quart;
  id trt;
run;

data &out.;
set &out.;
length stat $200 r1-r3 $ 30;;
if lowercase(varname)='n' then do; statord=1; stat='n'; end;
if lowercase(varname)='miss' then do; statord=2; stat='Missing, n (%)'; end;
if lowercase(varname)='bl' then do; statord=3; stat='BLOQ, n (%)'; end;
if lowercase(varname)='median' then do; statord=7; stat='Median'; end;
if lowercase(varname)='quart' then do; statord=8; stat='Q25, Q75'; end;
if lowercase(varname)='minmax' then do; statord=9; stat='Min, Max'; end;
if lowercase(varname)='meansd' then do; statord=10; stat='Mean (SD)'; end;
if lowercase(varname)='aci' then do; statord=11; stat='95% CI of Mean'; end;
if r_1 ne '' then r1=r_1;
if r_2 ne '' then r2=r_2;
if r_3 ne '' then r3=r_3;

drop r_1 r_2 r_3;
if statord=3 then do;
  if r1='' then r1='0';
  if r2='' then r2='0';
  if r3='' then r3='0';
end;

if statord=2 then do;
  if r1='' then r1='0';
  if r2='' then r2='0';
  if r3='' then r3='0';
end;

run;
%mend mmeans;

%mmeans(dsn=adbx, class=paramn PARAMCD PARAM avisitn avisit atptn atpt, var=aval, out=out1);
%mmeans(dsn=adbx, class=paramn PARAMCD PARAM avisitn avisit atptn atpt, var=pchg, out=out2);

*****;
* macro for Geometric Mean per mock ;
*****;
%macro mmeans(dsn=, class=, var=, out=);
*if aval=0 then present as NC for that treatment arm;
proc sql noprint;
  select count (distinct usubjid) into :aval1 from adbx where aval=0 and trt=1;
  select count (distinct usubjid) into :aval2 from adbx where aval=0 and trt=2;
  select count (distinct usubjid) into :aval3 from adbx where aval=0 and trt=3;
quit;

%put &aval1 &aval2 &aval3;
*all treatment have aval=0;

proc means data=&dsn. noprint nway;
  var &var.;
  class &class. trt;
  output out=results02 mean=mean std=std1 lclm=lci1 uclm=uci1;
run;

data results03;
set results02;
gmean1=exp(mean);
gmean=left(compress(put(round(gmean1,0.01), 11.2))), 11.2));
gcv=compress(put(0.001*ceil((sqrt(exp(std1*std1)-1)*100)/0.001),11.3));
if lci1 ne . then glci=exp(lci1);
if uci1 ne . then guci=exp(uci1);
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(0.01*floor(glci/0.01),11.2)) || ', ' || strip(put(0.01*ceil(gu
ci/0.01),11.2)));
run;

data &out._f1(keep=param paramn paramcd gmean gcv gmean1 glci guci gmeancv ci avisitn trt avisit ) ;
  set results03;

```

```

run;

proc transpose data=results03 out=&out prefix=r_ name=varname;
    by &class.;
    var gmeancv ci;
    id trt;
run;

data &out.;
    set &out.;
    length stat $200 r1-r3 $ 30;
    if lowercase(varname)='gmeancv' then do; statord=5; stat='Geometric Mean (CV%)'; end;
    if lowercase(varname)='ci' then do; statord=6; stat='95% CI of Geometric Mean'; end;
    if r_1 ne '' then r1=r_1;
    if r_2 ne '' then r2=r_2;
    if r_3 ne '' then r3=r_3;

    drop r_1 r_2 r_3;

run;

%mend mmeans;

%mmeans(dsn=adbx, class=paramn PARAMCD PARAM avisitn avisit atptn atpt, var=logaval, out=out3);

```

```

data tflds.t_15_02_04_66_02_f;
    set out3_f1;
    if trt=1 then trtpn=4;
    else if trt=2 then trtpn=5;
    else if trt=3 then trtpn=3;
run;
*check data for BLOQ/ALQ ;
proc freq data=adbx;
    table AVALC;
run;

```

```

*****;
* set together ;
*****;
data final1;
    set out1 out3;
run;

```

```

proc sort data=final1 out=final2;
    by paramn PARAMCD PARAM avisitn avisit atptn atpt statord stat varname;
run;

```

```

data out2d;
    set out2;
    if avisit="Baseline" then delete;
run;

```

```

proc sort data=out2d out=out2_s;
    by paramn PARAMCD PARAM avisitn avisit atptn atpt statord stat varname;
run;

```

```

data final;
    retain PARAMCD PARAM avisitn avisit atptn statord tp stat r1 c1 r2 c2 r3 c3 statord;
    merge final2(in=a) out2_s(in=b rename=(r1=c1 r2=c2 r3=c3));
    by paramn PARAMCD PARAM avisitn avisit atptn atpt statord stat varname;
    if a;

```

```

tp=avisit;

```

```

***should delete <missing, n(%)> if no missing n;

```

```

***4/22/2015 email from Jonh:

```

```

sometimes tables will have a value in brackets < > e.g., <Missing> or <BLOQ> as shown above.

```

```

This means to only report that value (after removing the brackets) in the table if there are such values to report.

```

```

For example, if there are 40 people in a treatment group and 30 are males and 9 are females then <Missing> would have 1.

```

```

However, if there are no missing values for sex then <Missing> should not be reported at all. ***;

```

```

if stat="Missing, n(%)" and r1="" and r2="" and r3="" and c1="" and c2="" and c3="" then delete;

```

```

if stat='95% CI of Mean' then do;
    if r1='' then r1='NC, NC';
    if r2='' then r2='NC, NC';

```

```

    if r3='' then r3='NC, NC';
    if tp^='Baseline' then do;
        if r1='' then r1='NC, NC';
        if r2='' then r2='NC, NC';
        if r3='' then r3='NC, NC';

        if c1='' then c1='NC, NC';
        if c2='' then c2='NC, NC';
        if c3='' then c3='NC, NC';
    end;
end;

if stat='95% CI of Geometric Mean' then do;
    if r1='' then r1='NC, NC';
    if r2='' then r2='NC, NC';
    if r3='' then r3='NC, NC';
    if tp^='Baseline' then do;
        if r1='' then r1='NC, NC';
        if r2='' then r2='NC, NC';
        if r3='' then r3='NC, NC';
    end;
end;

if stat='Geometric Mean (CV%)' then do;
    if r1='' then r1='NC, NC';
    if r2='' then r2='NC, NC';
    if r3='' then r3='NC, NC';
    if tp^='Baseline' then do;

        if r1='' then r1='NC, NC';
        if r2='' then r2='NC, NC';
        if r3='' then r3='NC, NC';
    end;
end;
keep paramn PARAMCD PARAM avisitn avisit atptn statord tp stat r1 c1 r2 c2 r3 c3;
run;

*****;
*create new page for each avisit for report ;
*****;

proc sql;
    create table page as
    select distinct paramn, PARAMCD, PARAM, avisitn, tp
    from final
    order by paramn, PARAMCD , PARAM, avisitn, tp;
quit;

data page1;
    set page;
    by paramn PARAMCD PARAM avisitn tp;
    if _n_ = 0 then page = 0;
    page + 1;
run;

proc sql;
    create table final_page as
    select distinct a.*, b.page
    from final as a
    left join page1 as b
    on a.avisitn=b.avisitn and a.tp = b.tp and a.paramn=b.paramn and a.paramcd=b.paramcd and a.param=b.param
    order by paramn, PARAMCD , PARAM, page, avisitn, avisit, atptn, statord;
quit;

data final_page(rename=(r1=THSm c1=THSm_chg r2=mCC c2=mCC_chg r3=SA c3=SA_chg));
    set final_page end=last;
    by page avisitn atptn statord;
    if last then call symputx("page", page);
        if statord=3 then do;
            c1='';
            c2='';
            c3='';
        end;

        if r1='' and r2='' and r3='' and c1='' and c2='' and c3='' then delete;
        if stat='BLOQ, n (%)' and r1='0' and r2='0' and r3='0' then delete;

run;

```

```

%let tflno=T_15_02_04_66_02;
data tfls.&tflno(keep=paramn paramcd param avisitn avisit tp stat THSm mCC SA THSm_chg mCC_chg SA_chg statord page);
  set final_page;
run;

*****;
*create output report ;
*****;

options number nodate orientation=landscape missing=' ';
ods escapechar='$';
%let linetop = \brdrt\brdrs\brdrw30;
%let linebot = \brdrb\brdrs\brdrw30;

%macro outrtf(blankn=130, halfblnk=N, dsn=);
%let title1 = %str(Table 15.2.4.66.2 Descriptive Statistics of Oxysterol Parameters - FAS);

%let TFL_Part=%scan(&_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;

%if &halfblnk=N %then %let halfblnk=;
%else %if &halfblnk=Y %then %let halfblnk=\~;

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=14
40 footery=1440 ;
ods noproctitle;
%do i=1 %to &page;
  title ;
  footnote;
  %let wd=0;
  ods proclabel = ' ';

data comp;
  set final_page end=eof;
  where page=&i;

  _firtitl="%&title1.";
  _upcas=(length("Path: &TFLpath.")-length(compress("Path:&TFLpath.", 'ABCDEFGHIJKLMNOPQRSTUVWXYZ')))/2;
  len=&blankn.-length("(page &i of &page)");

  if eof then do;
    call symput('_FSRTITL', trim(left(_firtitl)));
    call symput('_blankn', compress(put(len,best.)));
    call symput('param', strip(param));
  end;
  drop _firtitl _upcas len;
run;

ods listing close;

proc report data = comp headline headskip nowd split = '$' %if &i=1 %then %do; contents=' ' %end; %else %do; contents='' %end;;
  column tp stat ("THSm2.2$(N=&trt1)$&linebot" THSm THSm_chg )
           ("mCC$(N=&trt2)$&linebot" mCC mCC_chg)
           ("SA$(N=&trt3)$&linebot" SA SA_chg);
  define tp          /"Timepoint" order order=internal style={just=left cellwidth=0.9cm} style(header)={just=left} ;
  define stat        /"Statistic" display style={just=left cellwidth=1.9cm} style(header)={just=left} ;
  define THSm        /"Value" display style={just=c cellwidth=1.2cm} style(header)={just=center} ;
  define mCC         /"Value" display style={just=c cellwidth=1.2cm} style(header)={just=center} ;
  define SA          /"Value" display style={just=c cellwidth=1.2cm} style(header)={just=center};
  define THSm_chg    /"% Change(*)" display style={JUST=c cellwidth=1cm} style(header)={just=center};
  define mCC_chg     /"% Change(*)" display style={just=c cellwidth=1cm} style(header)={just=center};
  define SA_chg      /"% Change(*)" display style={just=c cellwidth=1cm} style(header)={just=center};
  compute after tp;
  line " ";
  endcomp;

```

```

compute before _page_ / style={just=left protectspecialchars=off};
  line "\b\fs24\sa24&_FSRTITL." ;
line " ";
line "Parameter (units): &param";
line " ";
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: * % Change from baseline, where baseline is defined as the last assessment prior to first randomized product use in
mCC / THSm2.2 arms or the last assessment prior to 10AM on Day 1 in the SA arm.';
  line "Note: 'Missing' percentages are based on the number of subjects indicated in the column header (N), while 'BLOQ' percentag
es are based on the number of subjects being summarized (n).";
  line " ";
  line 'Appendix 15.3.6.22';
  line "Study ID: ZRHM-REXA-08-US      Program: &TFLprg      Status: &status" &_blankn.*"\-\" "&sysdate" &_blankn.*"\-\" "(Page &i
of &page)";
  endcomp;
run;
%end;
ods rtf close;
ods results on;
ods path sashelp.tmplmst (read);
%amend outrtf;

%outrtf(blankn=36, halfblnk=N);
ods listing;

%m_logchk2;

```