

/*=====

*Covance Study ID : 000000106331

*Program Name : t_cohb_fas.sas

*Purpose : Descriptive Statistics of Blood COHb (%) - FAS

Table 15.2.4.1.2

*Input Data : adam.adsl, ADAM.adbx

*Output Data : tflds.T_15_02_04_01_02, tflds.T_15_02_04_01_02_F

*Macros Called : %m_printto, %m_logchk, %mmeans, %outrtf

*Programmed by : L.Ma

*Creation Date : 2015-05-14

*=====

*Modification History

*Date Initials No. Reason;

=====/

options notes nosource;

proc datasets lib=work nolist memtype=data kill; quit;

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

ods _all_ close;

ods listing;

%m_printto;

*=====

*** Creating dataset for figures ***

*=====;

/*Bring in data from ADBX for Blood COHb - fas Set per Mock*/

data adbx1_f;

set adam.adbx;

where anl02fl='Y' AND PARCAT1 = 'BIOMARKERS' AND PARAM = 'COHb (%)' AND LBSPEC =
'BLOOD'

AND (fasfl='Y');

run;

data adbx_f;

set adbx1_f;

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

if avisit not in ('Baseline' 'Day 1' 'Day 2' 'Day 3' 'Day 4' 'Day 5' 'Day 30' 'Day 60' 'Day 90') then
delete;

if aval ne 0 and aval ne . then logaval=log(aval);

run;

proc means data=adbx_f noprint nway;

var logaval;

class param paramn paramcd avalu trtpn trtp avisitn avisit atptn atpt ;

output out=rs_f mean=mean1 lclm=lci1 uclm=uci1;

run;

data rs_f;

```

set rs_f;

gmean1=exp(mean1);

mean=round(gmean1,0.01);

glci=exp(lci1);

guci=exp(uci1);

if not missing(glci) then lclm=0.01*floor(glci/0.01);

if not missing(guci) then uclm=0.01*ceil(guci/0.01);


/*According to QCer/Figure programmer, Keep Baseline data only one (when atpt=DAY 0 - 20:00
- 21:30). */

if avisit='Baseline' and atpt='DAY -1 - 20:00 - 21:30' then delete;


keep param paramn paramcd avalu trtpn trtp avisitn avisit atptn atpt lclm uclm mean;

run;


/*output data for figure; */

data tflds.T_15_02_04_01_02_F(keep=param paramn paramcd avalu trtpn trtp avisitn avisit atptn atpt
lclm uclm mean);

set rs_f;

run;


*=====

*** for rtf Table 15.2.4.1.2 ***

*=====;

```

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

```
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;
```

```
/*Bring in data from ADBX for Blood COHb - fas Set per Mock*/
```

```
data adbx1;
```

```
    set adam.adbx;
```

```
    where anl02fl='Y' AND PARCAT1 = 'BIOMARKERS' AND PARAM = 'COHb (%)' AND LBSPEC =  
'BLOOD'
```

```
    AND (fasfl='Y');
```

```
    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 avisit='Baseline';

    if avisit not in ('Baseline' 'Day 1' 'Day 2' 'Day 3' 'Day 4' 'Day 5' 'Day 30' 'Day 60' 'Day 90') then
delete;

    /*Keep Baseline data in one obs. per 5/15/2015 JH email*/

    if avisit='Baseline' then do; avisitn=100; atpt="DAY 0 - 20:00 - 21:30"; atptn=1.45; end;

    if aval ne 0 and aval ne . 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;

    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=$20.

        minmax length=$20.

        n    length=$20.

        miss length=$20.

        median length=$20.

        quart aci length=$20.;

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

    *for <missing, n(%)>;

    if trt=1 then do;

                                *format missing cell to 0 per John email on 8-5-2015;

                                if &trt1.=n1 then miss="0";

                                else miss=strip(put((&trt1.-n1), 8.)) || ' (' || strip(put(((&trt1.-
n1)*100)/&trt1., 8.1)) || ")";

                                end;

    else if trt=2 then do;

                                if &trt2.=n1 then miss="0";

                                else miss=strip(put((&trt2.-n1), 8.)) || ' (' || strip(put(((&trt2.-
n1)*100)/&trt2., 8.1)) || ")";

                                end;

    else if trt=3 then do;

                                if &trt3.=n1 then miss="0";

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

```

```

end;

if not missing(median1) then median = left(compress(put(round(median1,0.01),8.2)));

if not missing(mean1) and not missing(std1) then meansd =
left(compress(put(round(mean1,0.01),8.2))) || ' (' || left(compress(put(0.001*ceil(std1/0.001),8.3))) ||
');

if not missing(min1) and not missing(max1) then minmax = left(compress(put(min1,8.1))) || ', ' ||
left(compress(put(max1,8.1)));

if not missing(lci1) and not missing(uci1) then aci = strip(put(0.01*floor(lci1/0.01),8.2)) || ', ' ||
strip(put(0.01*ceil(uci1/0.01),8.2));

if not missing(q1) and not missing(q3) then quart = strip(strip(put(round(q1, 0.01),8.2)) || ', ' ||
strip(put(round(q3, 0.01),8.2)));

drop n1 mean1 std1 median1 min1 max1 q1 q3 uci1 lci1 ;

run;

proc transpose data=results03 out=&out. prefix=r name=varname;

by &class.;

var n miss meansd median minmax aci quart;

id trt;

run;

data &out.;

set &out.;

length stat $200;

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

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

if upcase(varname)='MEDIAN' then do; statord=7; stat='Median'; end;

```

```

        if upcase(varname)='QUART'    then do; statord=8; stat='Q25, Q75'; end;

        if upcase(varname)='MINMAX' then do; statord=9; stat='Min, Max'; end;

        if upcase(varname)='MEANSD' then do; statord=10; stat='Mean (SD)'; end;

        if upcase(varname)='ACI'      then do; statord=11; stat='95% CI of Mean'; end;

run;

%mend mmeans;

%mmeans(dsn=adbx, class=avisitn avisit atptn atpt, var=aval, out=out1);

%mmeans(dsn=adbx, class=avisitn avisit atptn atpt, var=pchg, out=out2);

*****
;

* macro for Geometric Mean per mock ;

*****
;

%macro mmeans(dsn=, class=, var=, out=);

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), 8.2)));

gcv=compress(put(0.001*ceil((sqrt(exp(std1*std1)-1)*100)/0.001),8.3));

glci=exp(lci1);

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),8.2)) || ' , '
|| strip(put(0.01*ceil(guci/0.01),8.2)));

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;

if upcase(varname)='GMEANCV' then do; statord=5; stat='Geometric Mean (CV%)'; end;

if upcase(varname)='CI' then do; statord=6; stat='95% CI of Geometric Mean'; end;

run;

%mend mmeans;

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

```

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

```
* BLOQ/ALOQ ;
```

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

*per Jh email on Wed 7/1/2015 9:34 AM ---- we only need to present BLOQ as a line item if there are BLOQ values for a given parameter/timepoint;

```
proc freq data=adam.adbx(where=(anl02fl='Y' AND PARCAT1 = 'BIOMARKERS' AND PARAM = 'COHb (%)'
AND LBSPEC = 'BLOOD' AND AQLFL='Y' ));
```

```
table AVALC;
```

```
run;
```

*no data for BLOQ values so no need to present BLOQ n(%) row.;

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

```
* set together ;
```

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

```
data final1;
```

```
set out1 out3;
```

```
run;
```

```
proc sort data=final1 out=final2;
```

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

```
run;
```

```
proc sort data=out2 out=out2_t;

    by avisitn avisit atptn atpt statord stat varname;

run;
```

```
data out2t;

    set out2_t;

    if avisitn=100 then delete;

run;
```

```
data final;

/*    retain avisitn avisit atptn statord tp stat r1 c1 r2 c2 r3 c3 statord;*/

    merge final2(in=a) out2t(in=b rename=(r1=c1 r2=c2 r3=c3));

    by avisitn avisit atptn atpt statord stat varname;

    if a;

    *format avisit/tp per mock;

    tp=avisit;

    if avisit="Day 5" then do; /*follow JH email on Tue 4/28/2015 10:50 AM*/

        tp=atpt;

        if index(tp,'WITHIN 15 MIN PRIOR TO SMOKING') then tp=tranwrd(tp,'WITHIN 15 MIN
PRIOR TO SMOKING','Within 15 Min Prior To Smoking'); /*per client comments on Tue 6/30/2015*/

        if index(tp,'DAY 5 -') then tp=tranwrd(tp,'DAY 5 -','Day 5, ');

    end;

    ***delete <missing, n(%)> if no missing n data for all;
```

```
    if stat="Missing, n (%)" and r1="0" and r2="0" and r3="0" and c1="0" and c2="0" and c3="0"
then delete;
```

```
    else if stat="Missing, n (%)" and tp="Baseline" and r1="0" and r2="0" and r3="0" then delete;
```

```
    else if stat="Missing, n (%)" and avisit="Day 5" and index(tp, "Within 15 Min Prior To
Smoking")>0 and r1="0" and r2="0" and c1="0" and c2="0" and r3="" and c3="" then delete;
```

```
    else if stat="Missing, n (%)" and avisit="Day 5" and index(tp, "08:00 - 09:30")>0 and r1="" and
r2="" and c1="" and c2="" and r3="0" and c3="0" then delete;
```

```
    keep 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 avisitn, atptn, tp
```

```
        from final
```

```
        order by avisitn, atptn, tp;
```

```
quit;
```

```
data page1;
```

```
    set page;
```

```
    by avisitn atptn 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.atptn = b.atptn and a.tp = b.tp

        order by 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);

run;


/*output report data; */

%let tflno=T_15_02_04_01_02;

data tflds.&tflno(keep=avisitn avisit tp stat THSm mCC SA THSm_chg mCC_chg SA_chg page);

    set final_page;

run;


*****.

*create output report ;

```

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

```
options number nodate orientation=landscape 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;
```

```
%macro outrtf(blankn=130, halfblnk=N);
```

```
%let title1 = %NRBQUOTE(Table 15.2.4.1.2 Descriptive Statistics of Blood COHb (%) - 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=1440 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;

        /* Amend title as needed */

        _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.)));

        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=1.0cm}  
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=1.1cm}  
style(header)={just=center} ;
```

```
define mCC          /"Value" display style={just=c cellwidth=1.1cm}  
style(header)={just=center} ;
```

```
define SA          /"Value" display style={just=c cellwidth=1.1cm}  
style(header)={just=center};
```

```
define THSm_chg      /"% Change(*)" display style={JUST=c cellwidth=1.2cm}  
style(header)={just=center};
```

```
define mCC_chg      /"% Change(*)" display style={just=c cellwidth=1.2cm}  
style(header)={just=center};
```

```
define SA_chg      /"% Change(*)" display style={just=c cellwidth=1.2cm}  
style(header)={just=center};
```

```
compute after tp;
```

```
line " ";
```

```
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: * % 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 last assessment
prior to 10AM on Day 1 in the SA arm.';

line ' ';

line 'Appendix 15.3.3.2';

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);

%mend outrtf;

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

```

```
ods listing;
```

```
%m_logchk;
```

```
/***** END OF FILE t_cohb_fas.sas *****/
```