

```

*****
* This sas program calculates CDC percentiles and z-scores based on
* the 2000 cdc growth charts
* (http://www.cdc.gov/growthcharts/cdc_charts.htm.
* The reference population is children examined in NCHS studies from 1963-65
to 1988-94.

* The program also calculates extended BMIz and extended BMI percentile based
on
* modelling a half-normal distribution for BMIs >= 95th percentile. These
calculations
* are based on children with obesity in the original growth charts data and
children with
* obesity in more recent NHANES data (though 2015-16)

* Biologically implausible values are flagged, but these values are not
necessarily incorrect.
* Please see the information * on 'Extreme or biologically implausible
values' on the web page.

* This file is meant to be called with a %include statement from your SAS
program.
* To replicate estimates from NHANES, BMI should be rounded to 0.1
*****;

*****;
***** macros for calculations *****;
*****;

/* comment out next line - used for SAS University Edition */
/* libname rekdir '/folders/myfolders/cdc_growth_charts/data'; */

%macro _zscore(var,l, m, s, z, p, f);
  if &var >0 then do;
    if abs(&L) ge 0.01 then &z=((&var / &M)**&L-1)/(&L * &S);
    else if .z < abs(&L) < 0.01 then &z=log(&var / &M) / &S;
    &p=probnorm(&z)*100;

    sdl=((&M - &M*(1 - 2 * &L * &S)**(1 / &L)) / 2);
    sdh=((&M * (1 + 2 * &L * &S)**(1 / &L) - &M) / 2);
    if &var lt &M then &f= (&var - &M) / sdl; else &f= (&var -
&M) / sdh;
  end;
%mend _zscore;

%macro _cuts(var,out,l,u);
  if &L <= &var <= &u then &out=0;
  else if &var > &u then &out=1; else if .< &var < &L then &out= -1;
%mend _cuts;

***** End of Macros *****;

data _orig_mydata_old; set mydata; _id=_n_; length agemos 8 sex 3;
  /* length and stand_ht are used later in deciding to use
  wt-for-stature or wt-for-len reference; */
  if (. < agemos < 24) then do; length=height; end;
  if agemos >=24 then stand_ht=height;

```

```

if 24 <= agemos < 240 then output _mydata; else output _old;
output _orig;

data _cinage _cinlen _cinht; set _mydata;
if agemos ge 0 and agemos lt 0.5 then _agecat=0;
  else _agecat=int(agemos+0.5)-0.5;
/*if bmi < 0 & ( weight>0 & height >0 & agemos >=24) then
bmi=weight/(height/100)**2;*/
output _cinage;

if length > . then do;
  if length >= 45 then _htcat=int(length+0.5)-0.5;
    if 45 <= length < 45.5 then _htcat=45;
  output _cinlen;
  end;
if stand_ht > . then do;
  if stand_ht ge 77.5 then _htcat=int(stand_ht+0.5)-0.5;
    else if 77<= stand_ht < 77.5 then _htcat=77;
  output _cinht;
  end;

*****;
** begin the for-age calcs - note that this calls up the refdir libname;
data crefage; set refdir.CDCref_d; where denom='age';
  /* contains all merged LMS data - use 'denom' variable; */
  length sex 3;
  proc sort; by sex _agecat;      proc sort data=_cinage; by sex _agecat;

data finfage; merge _cinage (in=a) crefage (in=b); by sex _agecat; if a;
  ageint = _agemos2-_agemos1; dage=agemos- _agemos1;

array l0 _llg _mlg _slg _lht _mht _sht _lwt _mwt _swt _lhc _mhc
  _shc
  _lbmi _mbmi _sbmi;
array l1 _llg1 _mlg1 _slg1 _lht1 _mht1 _sht1 _lwt1 _mwt1 _swt1 _lhc1 _mhc1
  _shc1
  _lbmi1 _mbmi1 _sbmi1;
array l2 _llg2 _mlg2 _slg2 _lht2 _mht2 _sht2 _lwt2 _mwt2 _swt2 _lhc2 _mhc2
  _shc2
  _lbmi2 _mbmi2 _sbmi2;
do over l0; l0= l1 + (dage * (l2 - l1)) / ageint; end;

if agemos < 24 then _mbmi=.; * theres a valid value for 23.5 months! ;

/* note that upper cutpoints were changed in 2016 to +4 (ht) and +8 (wt,
BMI); */

%_zscore(length, _llg, _mlg, _slg, lgz, lgpct, _Flenz);
  %_cuts(_flen, _bivlg, -5, 4);
%_zscore(stand_ht, _lht, _mht, _sht, stz, stpct, _Fstatz);
  %_cuts(_fstatz, _bivst, -5, 4);
%_zscore(weight, _lwt, _mwt, _swt, waz, wp, mod_waz);
  %_cuts(mod_waz, _bivwt, -5, 8);
%_zscore(bmi, _lbmi, _mbmi, _sbmi, bmiz, bmip, mod_bmiz);
  %_cuts(mod_bmiz, _bivbmi, -4, 8);

```

```

bmi50 = _mbmi * ((1 + _lbmi*_sbmi*probit(0.50))**(1/_lbmi)); * m *
((1+1*s*z)^(1/l));
bmi95 = _mbmi * ((1 + _lbmi*_sbmi*probit(0.95))**(1/_lbmi));
bmi95 = 100 * (bmi/bmi95); * % of 95th percentile;

*****;
*** calculations for extended BMIZ *****;
*****;
agey=agemos/12;
if sex=1 then sigma = 0.3728 + 0.5196*agey - 0.0091*agey**2;
    else if sex=2 then sigma = 0.8334 + 0.3712*agey - 0.0011*agey**2;

if bmip <= 95 then do; ext_bmiz=bmiz; ext_bmip=bmip; end;
if bmip > 95 then do;
    ext_bmip = 90 + 10 * (probnorm((bmi - bmi95) / sigma));
    if ext_bmip <= 99.999999999999992 then ext_bmiz = probit(ext_bmip/100);
    * if ext BMI percentile is too close to 100, the ext BMIZ can't be
calculated;
    * set these to 8.21 a few lines below;
end;

if bmi>=1.2*bmi95 & ext_bmiz < .z then ext_bmiz=8.21;

/*
*** other BMI metrics if desired;
data finfage; set finfage; * calcs for distance and %distance from median
PMID 31439056;
z1=((bmi/_mbmi) - 1) / _sbmi; * LMS formula when L=1;
if sex=1 & bmiz>.z then do; mref= 23.02029424; sref = 0.134539365; end;
if sex=2 & bmiz>.z then do; mref= 21.71699934; sref = 0.152974718; end;

dist1 = z1 * _mbmi * _sbmi; * unadj distance from median;
adj_dist1 = z1 * sref * mref; * adjusted (to age 20 y) dist from the median;
percl = z1 * 100 * _sbmi; * unadj %difference from median with L=1;
adj_percl = z1 * 100 * sref; * adj %difference from median with L=1;
*/

if bmiz > .z then do; obese=0; sev_obese=0; end;
if bmip >= 95 & bmiz >.z then obese=1;
if bmip95 >= 120 & bmiz > .z then sev_obese=1;

drop _llg1 _mlg1 _slg1 _lht1 _mht1 _sht1 _lwt1 _mwt1 _swt1
    _lbmi1 _mbmi1 _sbmi1 _llg2 _mlg2 _slg2 _lht2 _mht2 _sht2 _lwt2 _mwt2
    _swt2
    _lbmi2 _mbmi2 _sbmi2 _lwht1 _mwht1 _swht1 _lwht2
    _mwht2 _swht2 _lwlg1 _mwlg1 _swlg1 _lwlg2 _mwlg2 _swlg2;

*****;
*** begin for-length and for-stand_ht calcs;
*****;

*** begin for-length calcs, birth to 36 mos;
proc sort data=_cinlen; by sex _htcat;
data creflg; set reffdir.CDCref_d (keep=denom sex _lg1--_swlg2);
    where denom='length'; _htcat=_lg1; length sex 3;

```

```

proc sort data=creflg; by sex _htcat;

data finflg; merge _cinlen (in=a) creflg; by sex _htcat;
  if a & (43 < length <104);
  lenint = _lg2- _lg1; dlen=length - _lg1;
array l _lwl _mwl _swl;
array l1 _lwlg1 _mwlg1 _swlg1;
array l2 _lwlg2 _mwlg2 _swlg2;
do over l; l = l1 + (dlen * (l2 - l1)) / lenint; end;

%_zscore(weight, _lwl, _mwl, _swl, wlz, wlpct, _Fwlz);
  %_cuts(_Fwlz, _bivwlg, -4, 8);
keep _id sex _agecat agemos weight _Fwlz _bivwlg wlz wlpct;

*** begin for-stand_ht calcs, zwtstat.xls;
proc sort data=_cinht; by sex _htcat;
data crefht; set rekdir.CDCref_d
  (keep=denom sex _ht1 _ht2 _lwht1 _lwht2 _mwht1 _mwht2 _swht1 _swht2);
  where denom='height'; _htcat=_ht1; length sex 3;
  proc sort data=crefht; by sex _htcat;

data finfht; merge _cinht (in=a) crefht; by sex _htcat;
  if a & (77 < height <122);
  htint = _ht2- _ht1; dht=height - _ht1;
array l _lwh _mwh _swh;
array l1 _lwht1 _mwht1 _swht1;
array l2 _lwht2 _mwht2 _swht2;
do over l; l = l1 + (dht * (l2 - l1)) / htint; end;

%_zscore(weight, _lwh, _mwh, _swh, wstz, wstpct, _Fwstz);
  %_cuts(_Fwstz, _bivwst, -4, 8);
keep _id sex _agecat agemos weight _Fwstz _bivwst wstz wstpct;

*** combine the for-age, for-length, and for-height calcs;
proc sort data=finflg; by _id; proc sort data=finfht; by _id;
data lenht; merge finflg finfht; by _id;

proc sort data=finfage; by _id;
data _outdata; * define height vars as max of standing height and length
vars;
  merge finfage lenht; by _id;

  array a stz stpct _bivst _fstatz wstz wstpct _bivwst _fwstz ;
  array b lgz lgpct _bivlg _flenz wlz wlpct _bivwlg _fwlz;
  array c haz hp _bivht mod_haz whz whpct _bivwh mod_whz;
  do over c; if agemos ge 24 then c=a; else c=b; end;

  if .z < weight < 0.01 then do; waz=.; wp=.; bmiz=.; bmip=.; whz=.;
whpct=.; end;
  if .z < height < 0.01 then do; haz=.; hp=.; bmiz=.; bmip=.; whz=.;
whpct=.; end;
  if .z < headcir < 0.01 then do; headcz=.; headcpct=.; end;

  min= min(of _bivht _bivwt _bivbmi _bivhc _bivwh);
  if min>=0 then _bivlow=0; else if min= -1 then _bivlow=1;
  max= max(of _bivht _bivwt _bivbmi _bivhc _bivwh);

```

```

        if max=0 or max= -1 then _bivhigh=0; else if max= 1 then
        _bivhigh=1;

data _outdata; set _outdata;
keep mod_bmiz mod_haz _Flenz _Fstatz mod_waz mod_whz _fwstz _Fwlz
/* _bivbmi _bivht _bivlg _bivst _bivwh _bivwlg _bivwst _bivwt */
agemos agey bmi bmip bmip95 bmiz height haz stz lgz _id
lgpct sex stpct wp waz hp wstz wstpct _fstatz
sigma ext_bmiz ext_bmip obese sev_obese;

data _outdata; set _outdata _old;
* combine with older and younger (< 24 mo) excluded people;
label
    waz='weight-for-age Z'
    /*
    _bivbmi='BIV BMI-for-age'
    _bivht='BIV height-for-age'
    _bivwt='BIV weight-for-age'
    */
    mod_bmiz='modified BMI-for-age Z'
    mod_haz='modified height-for-age Z'
    mod_waz='modified weight-for-age Z'
    mod_whz='modified weight-for-height Z'
    /*
    bmi50 = 'CDC median BMI-for-age'
    bmi95 = 'CDC 95th pctl BMI-for-age'
    */
    bmip='BMI-for-age percentile'
    bmip95='% of 95th BMI percentile'
    bmiz='BMI-for-age Z'
    hp='height-for-age percentile'
    haz='height-for-age Z'
    wp='weight-for-age percentile'
    ext_bmiz='extended BMIz'
    ext_bmip='extended BMI percentile'
    sigma='scale param for ext BMI pctl/z-score'
    /*
    zl='LMS z-score when L=1'
    mref='reference value of M (240 mo)'
    sref='reference value of S (240 mo)'
    dist1='BMI distance from median'
    percl='BMI %distance from median'
    adj_dist1='Adjusted BMI distance from median'
    adj_percl='Adjusted BMI %distance from median'
    */
    agey='age in years'
    obese='BMI >= 95th pctl of growth charts'
    sev_obese='BMI >= 120% of 95th pctl'
;

proc sort data=_outdata; by _id; proc sort data=_orig; by _id;

data _cdcdata; update _outdata _orig; by _id;
* variables in _orig dataset will overwrite any changes that were made;
drop _id stz stpct _fstatz wstz wstpct _fwstz lgz lgpct _flenz
    _fwlz length stand_ht;

```