# **/\* The SAS program (HEI-2005 at the population/group level using NHANES 2001-2002 data (and MPED))**

# **Population Ratio Method.SAS \*/**

/\*This program creates component and total scores of the HEI-2005 for a population or a group. The 12 components include: Total Fruit, Whole Fruit, Total Vegetables, Dark Green and Orange Vegetables and Legumes, Total Grains, Whole Grains, Milk, Meat and Beans, Oils,

\*Saturated Fat, Sodium, and Calories from Solid Fat, Alcohol, and Added Sugar (SoFAAS). \*/

/\*Please see accompanying readme file. \*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**/\*INSTRUCTIONS – complete tasks 1-4 in this section, and run these SAS codes before proceeding to the HEI-2005 scoring program that follows\*/**

/\*1. Create a folder on your computer “home folder”, and save the FPED data, NHANES data, Demographic data, and the required HEI-2005 macros in it. Specify the path to the folder. \*/

%let home = C:\Users\Documents\MPED\_NHANES; /\*In this Example, the “home” folder is in C Drive, within Documents, and is called MPED\_NHANES. \*/

/\*2. Libnames and Filename here specify the input files. \*/

libname NH “&home\NH”;

libname MPED “&home\MPED”; /\*In this Example, NHANES data are in a folder called “NH”. MPED data are in a folder called “MPED”. These are SAS datasets. \*/

filename cnppadd “&home\cnppadd”; /\* The data for CNPP Addendum to MPED are “cnppadd”, in Excel format. All these are saved within the “home” folder\*/

/\*3. Create a folder in the "home" folder, where the output file, containing HEI-2005 component and total scores are to be exported. Specify the name of the folder. \*/

filename RES “&home\RES”; /\*In this Example, the folder is called “RES”, within the “home” folder, and the exported results will be a csv file called “result”. \*/

/\*4. Read in required HEI-2005 scoring macros. These macros must be saved within the home folder. \*/

%include “&home\hei2005.beanspeas.allocation.macro.sas”;

%include “&home\hei2005.score.macro.sas”;

/\*NOTE: Once you have completed all the steps above, all you need to do is run the SAS program below. Unless you used different names for your folders and datasets, no other action is required from you. \*/

TITLE 'HEI-2005 scores for NHANES 2001-2002, AGE >= 2, RELIABLE DIETS, Include Pregnant and Lactating Women';

/\*Section (I): Calculation of food group, nutrient, and energy intakes at the individual

participant level. \*/

/\*Step 1: locate the required datasets and variables\*/

**data** MPED;

set mped.MyPyrEquivDB\_v1;

foodcode=**1**\*drdifdcd;

\*delete 2 foodcodes and use values from the Addendum;

if foodcode in (**11710800**,**11710801**) then delete;

/\*In the HEI-2005, soy beverages are counted as part of the Milk component.

Convert the four soy beverage codes in the MPED from M\_SOY oz equivalents to D\_TOTAL cup

equivalents using the following conversion process based on FNDDS 1.0\*/

/\*FOODCODE=11310000, MILK, IMITATION, FLUID, SOY BASED (1 cup=244 grams)

FOODCODE=11320000, MILK, SOY, READY-TO-DRINK, NOT BABY (1 cup=245 grams)

FOODCODE=11321000, MILK, SOY, READY-TO-DRINK, NOT BABY'S, CHOCOLATE (1 cup=240 grams)

FOODCODE=11330000, MILK, SOY, DRY, RECONSTITUTED, NOT BABY (1 cup=245 grams) \*/

IF FOODCODE=**11310000** THEN DO;

M\_SOY=**0**;

D\_TOTAL=ROUND(**100**\*(**1**/**244**),**.001**);

END;

ELSE IF FOODCODE=**11320000** THEN DO;

M\_SOY=**0**;

D\_TOTAL=ROUND(**100**\*(**1**/**245**),**.001**);

END;

ELSE IF FOODCODE=**11321000** THEN DO;

M\_SOY=**0**;

D\_TOTAL=ROUND(**100**\*(**1**/**240**),**.001**);

END;

ELSE IF FOODCODE=**11330000** THEN DO;

M\_SOY=**0**;

D\_TOTAL=ROUND(**100**\*(**1**/**245**),**.001**);

END;

**run**;

**PROC** **SORT** DATA=MPED;

BY FOODCODE;

**run**;

**DATA** CNPPMPED\_WJFRT;

SET mped.cnppmypyrequivdb\_v1\_wjfrt;

**run**;

**PROC** **SORT** DATA=CNPPMPED\_WJFRT;

BY FOODCODE;

**run**;

\*get cnpp addendum version 2.0 B;

**proc** **import** datafile=cnppadd

out=cnppadd

dbms=xls

replace;

getnames=yes;

**run**;

**data** cnppadd;

set cnppadd;

if foodcode in (**11710800**,**11710801**);

**run**;

**PROC** **SORT** DATA=cnppadd;

BY FOODCODE;

**run**;

**DATA** Food;

SET nh.drxiff\_b;

FOODCODE=**1**\*DRDIFDCD; /\*convert variable name and type\*/

if DRDDRSTZ=**1**; /\*reliable dietary recall status\*/

**run**;

**PROC** **SORT** DATA=Food;

BY FOODCODE;

**run**;

/\*Use the individual's Day 1 dietary intake sample weight (WTDRD1) to account for

the survey sample design, nonresponse rate, and proportion of intake days

(on weekdays or weekend days). \*/

**DATA** Nutrient (keep=seqn WTDRD1 DRXTKCAL DRXTCARB DRXTSFAT DRXTALCO DRDTSODI);

SET nh.DRXTOT\_b;

if DRDDRSTZ=**1**; /\*reliable dietary recall status\*/

**run**;

**PROC** **SORT** DATA=Nutrient;

BY SEQN;

**run**;

**DATA** demo (keep=seqn RIDAGEYR RIAGENDR SDDSRVYR SDMVPSU SDMVSTRA);

SET nh.demo\_b;

**run**;

**PROC** **SORT** DATA=demo;

BY SEQN;

**run**;

/\*Step 2: Combine the required datasets, make the necessary exclusions,

and make necessary adjustments at the level of individual foods. \*/

**DATA** PYR;

MERGE MPED CNPPMPED\_WJFRT cnppadd;

BY FOODCODE;

**run**;

**DATA** FDPYR;

MERGE Food (IN=S) PYR (IN=N);

BY FOODCODE;

IF S AND N;

**run**;

/\*Convert NHANES 01-02 individuals' food intake amounts from grams to

MyPyramid equivalents\*/

**DATA** FDPYR;

SET FDPYR;

/\*Calculate intake for MyPyramid food groups\*/

ARRAY PYRVAR G\_TOTAL--A\_BEV WHOLEFRT FRTJUICE;

DO OVER PYRVAR;

PYRVAR=PYRVAR\*(DRXIGRMS/**100**);

END;

**run**;

**PROC** **SORT** DATA=FDPYR;

BY SEQN;

**run**;

/\*Calculate individual's food intake amounts for MyPyramid food groups in 1 day\*/

**PROC** **MEANS** DATA=FDPYR NOPRINT;

BY SEQN;

VAR G\_TOTAL--A\_BEV WHOLEFRT FRTJUICE;

OUTPUT OUT=PYRCALC SUM= ;

**run**;

/\*Individual's nutrient intake (DRXICARB, carbohydrate; DRXIALCO, alcohol) from beer, wine

and distilled spirits, but exclude cooking wine\*/

**DATA** BWLNUT (KEEP=SEQN DRXICARB DRXIALCO DRXILINE THREED);

SET food;

THREED=INT(FOODCODE/**100000**);

IF (**931** <= THREED <= **935**); /\*alcoholic beverages of beer, wine, and distilled spirits\*/

IF FOODCODE=**93401300** THEN DELETE; /\*Remove cooking wine\*/

**run**;

**PROC** **SORT** DATA=BWLNUT;

BY SEQN DRXILINE;

**run**;

/\*Identify Added Sugar intake from beer, wine

and distilled spirits, but exclude cooking wine\*/

**DATA** BWLPYR (KEEP=SEQN ADD\_SUG DRXIGRMS DRXILINE THREED);

SET FDPYR;

THREED=INT(FOODCODE/**100000**);

IF (**931** <= THREED <= **935**); /\*alcoholic beverages of beer, wine, and distilled spirits\*/

IF FOODCODE=**93401300** THEN DELETE; /\*Remove cooking wine\*/

**run**;

**PROC** **SORT** DATA=BWLPYR;

BY SEQN DRXILINE;

**run**;

**DATA** BEERWINELIQ;

MERGE BWLNUT BWLPYR;

BY SEQN DRXILINE;

**run**;

/\*Exclude calories from Added Sugars food group in alcoholic beverages\*/

**DATA** BEERWINELIQ;

SET BEERWINELIQ;

BY SEQN;

SUGGRAM=ADD\_SUG\***4**; /\*convert from teaspoons to grams of added sugars=grams of carbohydrate from added sugars\*/

NOSCARB=DRXICARB-SUGGRAM; /\*subtract grams of carbohydrate from added sugars from the total carbohydrate\*/

IF THREED IN (**931**,**932**,**934**) THEN BWCARBC=NOSCARB\***4**; /\*SoFAAS calories from carbohydrate in alcoholic beverages\*/

ETHCAL=DRXIALCO\***7**; /\*SoFAAS calories from alcohol (ethanol) in alcoholic beverages\*/

**run**;

/\*Calculate individual's SoFAAS calories from carbohydrate and alcohol (ethanol) in alcoholic beverages in 1 day\*/

**PROC** **MEANS** DATA=BEERWINELIQ NOPRINT;

BY SEQN;

VAR BWCARBC ETHCAL;

OUTPUT OUT=BEERWINELIQ SUM=BWCARBC ETHCAL;

**run**;

/\*Combine all required datasets\*/

/\*Only include individuals who are age 2 years and older\*/

**DATA** BOTH;

MERGE Nutrient (IN=F) BEERWINELIQ PYRCALC(IN=P) DEMO;

BY SEQN;

IF F AND P;

IF RIDAGEYR >= **2**; /\*individuals age 2 and older\*/

**run**;

/\*Step 3: Calculate NHANES 01-02 individual food and nutrient intakes for each individual. \*/

/\*Please note: MyPyramid equivalent values for total dairy intake (D\_TOTAL) in the HEI-2005 may be different from D\_TOTAL

in the MPED because soy beverages are counted as milk in the HEI-2005. \*/

**DATA** BOTH;

SET BOTH;

BY SEQN;

V\_DOL=V\_DPYEL+V\_DRKGR;

ALLMEAT=M\_MPF+M\_EGG+M\_NUTSD+M\_SOY;

/\*\*Calculate intake of Calories from SoFAAS\*\*/

/\*Calculate SoFAAS Calories from Added sugars, solid fat, and alcoholic beverages\*/

ADDSUGC=**16**\*ADD\_SUG; /\*calories from added sugars\*/

SOLFATC=DISCFAT\_SOL\***9**; /\*calories from solid fat\*/

IF ETHCAL < **0** THEN ETHCAL=**0**;

IF BWCARBC < **0** THEN BWCARBC=**0**;

EXFAAS=ADDSUGC+SOLFATC+ETHCAL+BWCARBC; /\*total SoFAAS calories as in kcal\*/

**run**;

/\*Section (II): Calculation of weighted means and a variance-covariance matrix and generation

of a Monte Carlo dataset, enabling standard errors to be calculated. \*/

/\*Step 1. Calculate the weighted means and the variance/covariance matrix for the dietary variables of interest. \*/

**data** one;

set both;

array comp (**14**) DRXTKCAL F\_TOTAL WHOLEFRT V\_TOTAL V\_DOL legumes G\_TOTAL g\_whl d\_total

allmeat DISCFAT\_OIL DRXTSFAT DRDTSODI EXFAAS;

\* Turn each variable into an observation of the single variable VBL,;

\* keeping track of the order with the dum\_num variable;

do i = **1** to **14**;

VBL = comp(i);

dum\_num = i;

output;

end;

**run**;

**data** one;

set one;

\* Create dummies with the same names as the original variables;

\* The i-th dummy gets a 1 if the observation is associated with the

\* i-th original variable;

array comp (**14**) DRXTKCAL F\_TOTAL WHOLEFRT V\_TOTAL V\_DOL legumes G\_TOTAL g\_whl d\_total

allmeat DISCFAT\_OIL DRXTSFAT DRDTSODI EXFAAS;

do i = **1** to **14**;

if dum\_num = i then comp(i) = **1**;

else comp(i) = **0**;

end;

drop i dum\_num;

**run**;

**proc** **surveyreg** data=one;

strata SDMVSTRA;

cluster SDMVPSU;

weight WTDRD1;

model VBL=DRXTKCAL F\_TOTAL WHOLEFRT V\_TOTAL V\_DOL legumes G\_TOTAL g\_whl d\_total

allmeat DISCFAT\_OIL DRXTSFAT DRDTSODI EXFAAS/noint covb;

\* Output the covariance matrix we wanted all along;

ods output covb=csd\_cov;

title2 "Tricking SURVEYREG into giving us the covariance matrix of means";

**run**;

**proc** **print** data=csd\_cov;

title2 "Printout of csd\_cov dataset -uses complex survey info";

**run**;

**proc** **means** data=both n min max mean;

weight wtdrd1;

var DRXTKCAL F\_TOTAL WHOLEFRT V\_TOTAL V\_DOL legumes G\_TOTAL g\_whl d\_total

allmeat DISCFAT\_OIL DRXTSFAT DRDTSODI EXFAAS;

title2 'look at weighted means';

output out=wtdm mean= ;

**run**;

**data** covdata (drop=Parameter);

set csd\_cov;

\_TYPE\_='COV ';

\_NAME\_=Parameter;

**run**;

**data** wtdm (drop=\_TYPE\_ \_FREQ\_);

set wtdm;

**run**;

**data** wtdm;

set wtdm;

\_TYPE\_='MEAN';

**run**;

**data** covdata;

set covdata wtdm;

**run**;

**proc** **print** data=covdata;

title2 'input to simnorml';

**run**;

/\*Step 2. In this step, a Monte Carlo data set with 10,000 rows is generated using the weighted means

and variance/covariance matrix from step 1\*/

**proc** **simnormal** data=covdata(type=cov) numreal=**10000** seed=**51230077** outseed out=sim\_data;

var DRXTKCAL F\_TOTAL WHOLEFRT V\_TOTAL V\_DOL legumes G\_TOTAL g\_whl d\_total

allmeat DISCFAT\_OIL DRXTSFAT DRDTSODI EXFAAS;

**run**;

**proc** **means** data=sim\_data n nmiss min max mean stddev;

var DRXTKCAL F\_TOTAL WHOLEFRT V\_TOTAL V\_DOL legumes G\_TOTAL g\_whl d\_total

allmeat DISCFAT\_OIL DRXTSFAT DRDTSODI EXFAAS;

title2 "Distributions of Simulated Data";

**run**;

**proc** **print** data=sim\_data(obs=**20**);

title2 "Listing of 20 Records from Simulated Data";

**run**;

/\*Section (III): Allocation of legumes (beans and peas) and application of the HEI-2005 scoring algorithm.\*/

/\*Step 1. Allocate legumes (beans and peas) using the Monte Carlo data set from Section II and the beans

and peas allocation macro. \*/

%***LEGALLOC*** (indat=sim\_data,

kcal=DRXTKCAL,

allmeat=allmeat,

v\_total=v\_total,

v\_dol=v\_dol,

legumes=legumes,

outdat=lsim\_data);

/\*Step 2. Apply the HEI-2005 scoring macro. \*/

%***HEI2005*** (indat=lsim\_data,

kcal=DRXTKCAL,

f\_total=f\_total,

wholefrt=wholefrt,

lv\_total=legume\_added\_v\_total,

lv\_dol=legume\_added\_v\_dol,

g\_total=g\_total,

g\_whl=g\_whl,

d\_total=d\_total,

lallmeat=legume\_added\_allmeat,

oil=DISCFAT\_OIL,

sfat=DRXTSFAT,

sodi=DRDTSODI,

exfaas=exfaas,

outdat=aftermac);

**run**;

**proc** **means** data=aftermac n nmiss min max mean stddev;

var legume\_added\_v\_total legume\_added\_v\_dol legume\_added\_allmeat frtden whfrden vegden dgvden grnden wgrnden dairyden

meatden oilden pctsfat sodden sofa\_perc;

title2 'after legume allocation and hei 2005 scoring macro';

**run**;

/\*Section (IV): Calculation of HEI-2005 component and total scores and their confidence intervals.\*/

/\*Step 1. This step uses univariate and means procedures to compute total and component scores and their standard errors. \*/

**proc** **univariate** data=aftermac noprint;

var HEI1\_TOTALFRUIT HEI2\_WHOLEFRUIT HEI3\_TOTALVEG HEI4\_DARKVEG HEI5\_TOTALGRAIN HEI6\_WHOLEGRAIN

HEI7\_MILK HEI8\_MEATBEAN HEI9\_OIL HEI10\_SATFAT HEI11\_SODIUM HEI12\_EXFAAS HEI2005\_TOTAL\_SCORE;

output out=ci pctlpts=**2.5** **97.5** pctlpre=h1\_ h2\_ h3\_ h4\_ h5\_ h6\_ h7\_ h8\_ h9\_ h10\_ h11\_ h12\_ totscore\_;

**run**;

**proc** **means** data=aftermac noprint;

var HEI1\_TOTALFRUIT HEI2\_WHOLEFRUIT HEI3\_TOTALVEG HEI4\_DARKVEG HEI5\_TOTALGRAIN HEI6\_WHOLEGRAIN

HEI7\_MILK HEI8\_MEATBEAN HEI9\_OIL HEI10\_SATFAT HEI11\_SODIUM HEI12\_EXFAAS HEI2005\_TOTAL\_SCORE;

output out=stat min=h1\_min h2\_min h3\_min h4\_min h5\_min h6\_min h7\_min h8\_min h9\_min h10\_min h11\_min h12\_min totscore\_min

max=h1\_max h2\_max h3\_max h4\_max h5\_max h6\_max h7\_max h8\_max h9\_max h10\_max h11\_max h12\_max totscore\_max

mean=h1\_mean h2\_mean h3\_mean h4\_mean h5\_mean h6\_mean h7\_mean h8\_mean h9\_mean h10\_mean h11\_mean h12\_mean totscore\_mean

stddev= h1\_stddev h2\_stddev h3\_stddev h4\_stddev h5\_stddev h6\_stddev h7\_stddev h8\_stddev h9\_stddev h10\_stddev h11\_stddev h12\_stddev totscore\_stddev;

**run**;

/\*Step 2. This step prepares the results for display\*/

**data** all;

merge ci stat;

**run**;

**data** result (keep=score slabel min max mean stderr low high);

set all;

score='HEI1 ';

slabel='HEI-2005 COMPONENT 1 TOTAL FRUIT';

min=h1\_min;

max=h1\_max;

mean=h1\_mean;

stderr=h1\_stddev;

low=h1\_2\_5;

high=h1\_97\_5;

output result;

score='HEI2';

slabel='HEI-2005 COMPONENT 2 WHOLE FRUIT';

min=h2\_min;

max=h2\_max;

mean=h2\_mean;

stderr=h2\_stddev;

low=h2\_2\_5;

high=h2\_97\_5;

output result;

score='HEI3';

slabel='HEI-2005 COMPONENT 3 TOTAL VEGETABLES';

min=h3\_min;

max=h3\_max;

mean=h3\_mean;

stderr=h3\_stddev;

low=h3\_2\_5;

high=h3\_97\_5;

output result;

score='HEI4';

slabel='HEI-2005 COMPONENT 4 DARK GREEN & ORANGE VEG & LEGUMES';

min=h4\_min;

max=h4\_max;

mean=h4\_mean;

stderr=h4\_stddev;

low=h4\_2\_5;

high=h4\_97\_5;

output result;

score='HEI5';

slabel='HEI-2005 COMPONENT 5 TOTAL GRAINS';

min=h5\_min;

max=h5\_max;

mean=h5\_mean;

stderr=h5\_stddev;

low=h5\_2\_5;

high=h5\_97\_5;

output result;

score='HEI6';

slabel='HEI-2005 COMPONENT 6 WHOLE GRAINS';

min=h6\_min;

max=h6\_max;

mean=h6\_mean;

stderr=h6\_stddev;

low=h6\_2\_5;

high=h6\_97\_5;

output result;

score='HEI7';

slabel='HEI-2005 COMPONENT 7 MILK';

min=h7\_min;

max=h7\_max;

mean=h7\_mean;

stderr=h7\_stddev;

low=h7\_2\_5;

high=h7\_97\_5;

output result;

score='HEI8';

slabel='HEI-2005 COMPONENT 8 MEAT & BEANS';

min=h8\_min;

max=h8\_max;

mean=h8\_mean;

stderr=h8\_stddev;

low=h8\_2\_5;

high=h8\_97\_5;

output result;

score='HEI9';

slabel='HEI-2005 COMPONENT 9 OILS';

min=h9\_min;

max=h9\_max;

mean=h9\_mean;

stderr=h9\_stddev;

low=h9\_2\_5;

high=h9\_97\_5;

output result;

score='HEI10';

slabel='HEI-2005 COMPONENT 10 SATURATED FAT';

min=h10\_min;

max=h10\_max;

mean=h10\_mean;

stderr=h10\_stddev;

low=h10\_2\_5;

high=h10\_97\_5;

output result;

score='HEI11';

slabel='HEI-2005 COMPONENT 11 SODIUM';

min=h11\_min;

max=h11\_max;

mean=h11\_mean;

stderr=h11\_stddev;

low=h11\_2\_5;

high=h11\_97\_5;

output result;

score='HEI12';

slabel='HEI-2005 COMPONENT 12 CALORIES FROM SOLID FAT, ALCOHOL & ADDED SUGAR (SoFAAS)';

min=h12\_min;

max=h12\_max;

mean=h12\_mean;

stderr=h12\_stddev;

low=h12\_2\_5;

high=h12\_97\_5;

output result;

score='TOTAL HEI 2005';

slabel='TOTAL HEI-2005 SCORE';

min=totscore\_min;

max=totscore\_max;

mean=totscore\_mean;

stderr=totscore\_stddev;

low=totscore\_2\_5;

high=totscore\_97\_5;

output result;

**run**;

/\*Step 3 - This step displays the results\*/

**proc** **print** data=result;

id score;

var slabel min max mean stderr low high;

title2 'complex survey design population method - mean and confidence interval of HEI-2005 using NH 01/02 data';

**run**;

/\*Step 4 - This step provides an option to export the results into a CSV file that can be opened in Excel.\*/

**proc** **export** data=result

file=res

dbms=csv

replace;

**run**;