The NCI Users' Guide for Analysis of Usual Intake Distributions:
For Use with the Mixtran Version 1.1 and Distrib Version 1.1 Macros
02/06/2008

The SAS programs **mixtran_macro_v1.1.sas** and **distrib_macro_v1.1.sas** contain the MIXTRAN macro and the DISTRIB macro, respectively. These two macros can be used to estimate the distribution of usual intake of a food or nutrient in a population, using reported intake from repeat 24-hour recalls. The macros can be used to estimate the distributions of episodically consumed foods/nutrients and foods/nutrients consumed (nearly) every day. The macros can perform a weighted analysis of survey data such as NHANES.

The MIXTRAN macro fits a nonlinear mixed effects model to the repeat 24-hour recalls, using the NLMixed Procedure. For episodically consumed foods/nutrients, the macro fits a two-part model that defines the distribution of reported intake on a given day as the probability of consumption × conditional distribution of amount consumed on a consumption day. For foods/nutrients consumed every day, the macro fits a one-part model of amount consumed.

The DISTRIB macro inputs the parameter estimates and predicted values from the mixed effects model estimated by the MIXTRAN macro and uses a Monte Carlo method to estimate the distribution of usual intake.

The user must perform additional calculations, such as balanced repeated replication (BRR), in order to obtain standard errors and confidence intervals for the percentiles and mean from the distribution of usual intake. BRR calculations require that the user writes a SAS program that calls the MIXTRAN macro and the DISTRIB macro using the appropriate weight for each replication. MIXTRAN has several options (e.g. start_val1, start_val2, start_val3, and vcontrol) that are helpful for a user that needs to make repeated calls to MIXTRAN and DISTRIB to obtain BRR standard errors. We plan to provide, in the future, an additional macro that can perform the BRR calculations.

Caution: Note that the standard errors and p-values output by the MIXTRAN macro are only valid for an unweighted analysis (i.e. analysis of a simple random sample). In a weighted analysis (e.g. analysis of NHANES data), calculation of these standard errors requires additional programming to implement a replication method such as BRR.

For a detailed description of the nonlinear mixed effects model and Monte Carlo method used in the macros, see:

Tooze JA, Midthune D, Dodd KW, Freedman LS, Krebs-Smith SM, Subar AF, Guenther PM, Carroll RJ, Kipnis V. A new statistical method for estimating the usual intake of episodically consumed foods with application to their distribution. *J Am Diet Assoc*. 2006;106:1575-1587.

To use the MIXTRAN and DISTRIB macros in a SAS program, insert the following %include statements into your program prior to calling the macros:

```
%include "/mypath /mixtran_macro_v1.1.sas" ;
%include "/mypath /distrib_macro_v1.1.sas" ;
```

where "/mypath/" is the path name to the directory in which the programs **mixtran_macro_v1.1.sas** and **distrib_macro_v1.1.sas** are stored.

The MIXTRAN Macro

In the same SAS program, following the statement **%include** "/mypath **/mixtran_macro_v1.1.sas";**

call the MIXTRAN macro using the syntax:

```
%macro mixtran(data=, response=, foodtype=, subject=, repeat=, covars_prob=, covars_amt=,
        outlib=, modeltype=, lambda=, replicate_var=, seq=, weekend=, vargroup=,
        numvargroups=, subgroup=, start_val1=, start_val2=, start_val3=,
        vcontrol=, nloptions=, titles=, printlevel=);.
```

An example of a call to the MIXTRAN macro is:

```
%mixtran (data=male, response=add_sug, foodtype=add_sug, subject=seqn,
      repeat=drddaycd, covars_prob=,
      covars_amt=agegrp4 agegrp5 agegrp6 agegrp7 agegrp8 race2 race3,
      outlib=mylib, modeltype=amount, lambda=, replicate_var=rndw1,
      seq=seq2, weekend=weekend, vargroup=, numvargroups=,
      subgroup=agegroup, start_val1=, start_val2=, start_val3=,
      vcontrol=, nloptions=qmax=61, titles=4, printlevel=2);
run;.
```

The following list provides an explanation of each of the parameters in the MIXTRAN macro.

data — **Mandatory.** The name of the SAS data set to be used in the analysis. Do NOT use the names "data" or "data0" which are reserved for the MIXTRAN macro. For a work data set named "nhanes" for example, the syntax would be "data=nhanes,". For a data set saved to disk the syntax might be "data=inlib.nhanes,". In our example given above, the data set to be analyzed is a work data set named "male", so the syntax is:

data=male,.

response — **Mandatory.** The name of the food or nutrient 24-hour recall variable. In our example, the variable being analyzed is called add_sug. The syntax is:

response=add_sug,.

foodtype — **Mandatory.** A character string used to name the parameter and predicted data sets output by the MIXTRAN macro. The value must be a string valid in a SAS data set name. (For an explanation of the data sets output by the

MIXTRAN macro see the section SAS Data Sets Saved by the MIXTRAN Macro.) This parameter is used to differentiate the output data sets from other calls to the MIXTRAN macro. In our example, the syntax is:

foodtype=add_sug,.

subject **Mandatory.** The name of the variable that uniquely identifies each subject (i.e. ID variable). In our example, the ID variable is the variable "seqn", so the syntax is:

subject=seqn,.

repeat **Mandatory.** The name of the variable that indexes repeated observations for each subject. This variable indexes which 24-hour recall appears on a given record for a subject, so the value on a record should be an integer value of 1 or more. In our example, the "drddaycd" variable equals 1 for the first 24-hour recall and equals 2 for the second 24-hour recall, so the syntax is:

repeat=drddaycd,.

covars_prob **Optional.** A list of covariates for the first part of the model that models the probability of consumption. Covariates must be separated by spaces. Interactions must be in the order specified by PROC GENMOD. In an amount-only model, this parameter should be left blank. In our example, we specify an "amount" model, so this parameter is left as a null string. The syntax is:

covars_prob=,.

covars_amt **Mandatory.** A list of covariates for the second part of the model that models the consumption-day amount. Covariates must be separated by spaces. Interactions must be in the order specified by PROC GENMOD. These covariates are used in all model types. In our example, the syntax is:

covars_amt=agegrp4 agegrp5 agegrp6 agegrp7 agegrp8 race2 race3,.

outlib **Mandatory.** A library name reference to the directory where the data sets output from the macro will be saved. This library name must be specified in the SAS program prior to calling the macro, for example:

LIBNAME mylib "/myfiles/data/";.

Then the syntax in the macro call is:

outlib=mylib,.

modeltype **Mandatory.** There are three model types available. The options for this parameter are:

CORR	correlated model.	All 3 portions of the macro will be executed.
NOCORR	uncorrelated model.	The probability and amount portions of the macro will be

executed, but not the correlated portion.

AMOUNT	amount-only model.	Only the amount portion of the macro will be executed.

In our example we specify an "amount" model. The syntax is:

modeltype=amount,.

lambda

Optional. User specified value for the Box-Cox transformation parameter, lambda. If no value is specified, a value will be calculated. Our example lets the macro calculate an appropriate lambda value, so the parameter is left as a null string. The syntax is:

lambda=,.

replicate_var

Optional. The name of a weight variable to be used in the REPLICATE statements of the three calls to the SAS NLMixed procedure. The specified variable must be integer valued. The same variable will be used in all SAS procedures where a FREQ or WEIGHT statement is appropriate and is used in the calculation of the distributions in the DISTRIB macro. This variable is usually referred to as a weight variable. In our example, we use a variable named "rndw1". It will also be used as part of the name of the distribution data set saved by the DISTRIB macro. For an explanation of data sets output by the DISTRIB macro, see the section SAS Data Sets Saved by the DISTRIB Macro. In our example the syntax is:

replicate_var=rndw1,.

seq

Optional. The name of one or more dummy variables indicating the sequence number of each record if a subject has more than one record. The number of seq variables is one less than the possible number of records per person. For example a data set with two records per person will have just one seq variable, set to 0 for the person's first record and set to 1 for the second record. In our example, a subject can have two records. The variable "drddaycd", specified in "repeat=drddaycd", has possible values of 1 or 2, so we coded a new variable, named seq2. The value of seq2 is 1 when drddaycd=2 and 0 when drddaycd=1. The syntax in our example is:

seq=seq2,.

Note: The Distrib macro estimates distributions of intake under the assumption that the value(s) of all dummy variables specified in the "seq=" parameter have value zero. This process adjusts the usual intake for time-in-sample or sequence effects often noted in analysis recall data, where reported intake on the first application of the recall is distributed differently from data collected on subsequent recalls. Alternative coding for the dummy variables representing sequence may be used, but the user must understand the consequences of using such coding given the behavior of the DISTRIB macro.

weekend

Optional. The name of a binary variable designating if a record is a weekday (Monday-Thursday) or a weekend (Friday-Sunday) record. The variable should be coded as 0 if a weekday or 1 if a weekend. No other codes are permitted. This variable can NOT also be named in the covariate parameters

as the program will automatically add the weekend variable to those lists. If a weekend variable is specified, then the "vargroup=" and "numvargroups=" parameters may also be utilized. In our example, we are doing a weekend run, using a variable named "weekend". The syntax is:

weekend=weekend,.

vargroup **Optional.** The name of a variable, coded as 1 or 2, which groups observations and allows the model to incorporate a separate residual variance parameter for each of these groups of observations. This parameter can **only** be specified if a variable has been specified for the "weekend=" parameter. If this parameter is specified, then both the "weekend=" and "numvargroups=" parameters must also be specified. Currently the macro only allows a weekend variable to be used. This parameter can **only** be specified if a variable has been specified for the "weekend=" parameter. In this case if the weekend variable value is 0, the vargroup variable value must be 2, and if the weekend variable value is 1, then the vargroup variable value must be 1. In our example, we are doing a weekend run without variance groups, so the syntax is:

vargroup=,.

numvargroups **Optional.*** The number of groups in the vargroup parameter above. Currently this value can only be 2. This parameter can **only** be specified if a variable has been specified for the "weekend=" parameter. * If this parameter is called then both the "weekend=" and "vargroup=" parameters must also be specified. In our example, we are doing a weekend run without variance groups, so the syntax is:

numvargroups=,.

subgroup **Optional.** The name of a single categorical variable to be used when calculating distributions of input by group. The subgroup variable is used in the DISTRIB macro, but must be passed through the MIXTRAN macro because the values are saved in the output data set of predicted values. Any variables used to create the subgroup variable MUST be among the covariates specified in "covars_prob" and/or "covars_amt". In our example we are using a subgroup variable named "agegroup", created from the ages of the subjects. The syntax is:

subgroup=agegroup,.

start_val1 **Optional.** The name of a data set with starting values for the first SAS NLMixed procedure (i.e. NLMixed for the probability model). This parameter can **only** be invoked when the MIXTRAN macro is being used in a rerun, that is, for runs that model the same data using different weights. For base runs this parameter is always a null string. The start_val1 data set is only created by a base run. The first set of starting values is used in NLMixed for the probability part of the model. The only values that can be input on a re-run are the parameter estimates saved from this NLMixed by a prior execution of the MIXTRAN macro. This data set always uses the name "_parmsf1", followed

by the value in the parameter "foodtype". It will have been saved to the directory specified in the base run using the parameter "outlib=". If we were doing a re-run in our example the syntax would be "start_val1= mylib._parmsf1_add_sug,". However, our example is a base run, so our syntax is simply "start_val1=,". The "_parmsf1" data set is only available following a base run for a correlated model (i.e. modeltype=CORR) or an uncorrelated model (i.e. modeltype=NOCORR). In our example the syntax is:

start_val1=,.

start_val2 **Optional.** The name of a data set with the starting values for the second SAS NLMixed procedure (the amount portion of the model). The same rules apply as for start_val1. This parameter can **only** be used on a re-run and uses the "_parmsf2" data set, created by a previous execution of the MIXTRAN macro. The "_parmsf2" data set is available for all model types. In our example, we are doing a base run. The syntax is:

start_val2=,.

start_val3 **Optional.** The name of a data set with the starting values for the third SAS NLMixed procedure (the correlated model). The same rules apply as for start_val1. This parameter can **only** be used on a re-run and uses the "_parmsf3" data set, created by a previous execution of the MIXTRAN macro. The "_parmsf3" data set is only available following a base run for a correlated model (i.e. modeltype=CORR). In our example, we are doing a base run. The syntax is:

start_val3=,.

vcontrol **Optional.*** A one to six character string used **only** when starting values are being supplied via the parameters start_val1, start_val2, or start_val3. This situation is the case we consider a "re-run". * In a re-run the vcontrol parameter is **mandatory**. It is used to version control the names of output data sets and as a flag within the macro to indicate that this run is a re-run. Since the supplied string is used as part of the name of SAS data sets output by the MIXTRAN macro, the string should only contain characters valid in SAS data set names. Our example is a base run, so this parameter is null. For our example, the syntax is:

vcontrol=,.

nloptions **Optional.** A list of options for the PROC NLMixed statement of the SAS NLMixed Procedure (see the SAS documentation for NLMixed for a list of options). Any options listed will be used in fitting the models. As an example, one could specify: **nloptions=technique=truereg gconv=1e-10 itdetails,.** In our example, we specify the option "qmax=61", so the syntax is:

nloptions=qmax=61,.

titles **Optional**. An integer from 0 to 4. Up to 4 title lines can be reserved for custom titles in the .lst output. If no number is supplied, the default is 0. If the

number is more than 4, it will be changed to 4. Our example reserves 4 title lines. The syntax is:

titles=4,.

printlevel **Optional.** An integer from 1 to 3. This parameter controls the amount of information printed to the .lst file.

1 prints only summary reports.

2 prints summary reports and the output from the NLMixed procedures. This value is the default.

3 prints summary reports and the output from all of the statistical procedures.

In our example, we opted for the default level of 2.
Since this parameter is the last parameter in the call to the MIXTRAN macro, it is not followed by a comma. The syntax is:

printlevel=2.

SAS Data Sets Saved by the MIXTRAN Macro

The MIXTRAN macro creates a number of SAS data sets that are saved to disk for later use. Each data set is saved to the directory named by the reference in the macro parameter "outlib=" as described above. After each execution of an NLMixed procedure, the parameter estimates from that procedure are saved as a data set. This data set can be used as starting input for later executions of the MIXTRAN macro. In addition data sets of the predicted values and the parameter estimates for both the correlated and the uncorrelated models are saved for possible use with the DISTRIB macro.

The data sets saved, for use as input in future re-runs of the MIXTRAN macro, are the tables of parameter estimates output by the SAS NLMixed procedure and an expression to calculate the predicted values. The data sets are named using the following conventions:

outlib._dsn_foodtype

where:

outlib is the library name given in the MIXTRAN parameter outlib thus depends on user input.

_dsn is one of the following names:

_parmsf1 This data set captures the parameter estimates output by the SAS NLMixed procedure for the probability model in a base run. It can be used as input for the starting values for this model in a re-run. In a re-run this data set is referenced in the MIXTRAN parameter "start_val1".

_parmsf2 This data set captures the parameter estimates output by the SAS NLMixed procedure for the amount model in a base run. It can be used as input for the starting values for this model in a re-run. In a re-run this data set is referenced in the MIXTRAN parameter "start_val2".

_parmsf3 This data set captures the parameter estimates output by the SAS NLMixed procedure for the correlated model in a base run. It can be used as input for the starting values for this model in a re-run. In a re-run this data set is referenced in the MIXTRAN parameter "start_val3".

etas This data set contains character strings that are interpreted by the MIXTRAN macro to calculate the predicted values. This data set is only output by a base run and only used in a re-run, and it is automatically utilized if the "vcontrol" parameter is in use.

_foodtype is obtained from the MIXTRAN macro's "foodtype" parameter thus depends on user input.

Several data sets are saved for use as input to the DISTRIB macro. The DISTRIB macro needs the predicted values for each subject and the parameter estimates calculated by the MIXTRAN macro. The predicted values and the parameter estimates are saved from the amount-only model, the uncorrelated model, and the correlated model for the AMOUNT, NOCORR, and CORR model types, respectively. Also, for the CORR model type, the predicted values and parameter estimates are also saved from the uncorrelated model which is fit in order to calculate starting values for the correlated model. The data sets are named using the following conventions:

outlib**._param_unc_**foodtype_vcontrol and
outlib**._pred_unc**_foodtype_vcontrol

where:

outlib is the library name given in the MIXTRAN parameter "outlib" thus depends on user input.

_param is used to indicate a parameter data set. This data set consists of one record and always includes "_param" in the data set name.

_pred is used to indicate a data set of predicted values. This data set consists of one record per person and always includes "_pred" in the data set name.

_unc is used to indicate the parameter and predicted data sets from the uncorrelated model and the amount-only model. The MIXTRAN macro will include the string "_unc" in these data set names. The MIXTRAN macro will not include the string "_unc" for the output data sets from the correlated model.

_foodtype is obtained from the MIXTRAN macro's "foodtype" parameter thus depends on user input.

_vcontrol is obtained from the MIXTRAN macro's "vcontrol" parameter thus depends on user input. Only re-runs will have this value appended to the data set name.

Thus, in our example which considers an "amount" model, the output data sets will be named:

mylib._parmsf2_add_sug
mylib.etas_add_sug

mylib._param_unc_add_sug
mylib._pred_unc_add_sug.

If we had fit a correlated model using a response variable named "fish" the output data sets would have been named:

mylib._parmsf1_fish
mylib._parmsf2_fish
mylib._parmsf3_fish
mylib.etas_fish

mylib._param_unc_fish
mylib._pred_unc_fish
mylib._param_fish
mylib._pred_fish.

Variable Names for the Covariates in the MIXTRAN macro

The variable names for the covariates used in the MIXTRAN macro are amended during the execution of the macro. This modification is performed in part because it is possible to use the same variables for both the amount and probability parts of the model, and these variables need to be differentiated during the macro processing. The new variable names also allow the macro to maintain the order of the variables as originally entered by the user. The covariates named in the "covars_amt" parameter will be modified, so the variable name is prefixed by the letter A, a sequence number, and an underscore, so the format is "Ann_". The first named covariate will be prefixed by the string "A02_", the second variable in the list will be prefixed by the string "A03_" and so on. In our example, "agegrp4" will become "A02_AGEGRP4", "agegrp5" will become "A03_AGEGRP5" etc. In a similar fashion, the covariates named in the parameter "covars_prob" for the probability part of the model will be prefixed using the format "Pnn_". The output from the SAS procedures and the saved data sets will reflect the new variable names. The intercept is always named A01_INTERCEPT or P01_INTERCEPT for the amount and probability intercepts, respectively.

The DISTRIB macro

The DISTRIB macro can be called in the same program as the MIXTRAN macro or in a later program. The DISTRIB macro can only be called after the MIXTRAN macro has been executed. The DISTRIB macro can only be successful if the MIXTRAN macro completed properly.

Call the DISTRIB macro using the syntax:

```
%macro Distrib (seed=, nsim_mc=, modeltype=, pred=, param=,
          outlib=, cutpoints=, ncutpnt=, byvar=, subgroup=,
          subject=, titles=, food=);.
```

An example of a call to the DISTRIB macro is:

```
%Distrib (seed=5454768, nsim_mc=100, modeltype=amount,
          pred=mylib._pred_unc_add_sug,
          param=mylib._param_unc_add_sug, outlib=mylib,
          cutpoints= 10 12 14  17  22 25 33 38 42 47, ncutpnt=10,
          byvar=, subgroup=agegroup, subject=seqn, titles=4, food=add_sug);
  run;.
```

The following list provides an explanation of each of the parameters in the DISTRIB macro.

seed — **Mandatory.** The seed for the random number generator used for the Monte Carlo simulation of the random effects u1 and u2. In our example given above, the syntax is:
seed=5454768,.

nsim_mc — **Mandatory.** The number of repetitions to be used in the Monte Carlo simulation. For each subject, one record will be output for each repetition. This number must be an integer. In our example, 100 repetitions are desired, so the syntax is:
nsim_mc=100,.

modeltype — **Mandatory.** The modeltype is usually the same model as used in the MIXTRAN macro that prepared the data for input to the DISTRIB macro. The available model types are:
CORR correlated
NOCORR uncorrelated
AMOUNT amount-only.
The declaration of the modeltype affects which parameter and predicted data sets are input to the DISTRIB macro. If the modeltype in the MIXTRAN macro was CORR and the correlated model was fit successfully, then the correlated data sets can be input to the DISTRIB macro. If the modeltype in the MIXTRAN macro was NOCORR or AMOUNT, then only the data sets

labeled with "_unc" can be input to the DISTRIB macro. Please see the section on SAS Data Sets Saved by the MIXTRAN Macro above for an explanation of data set naming conventions. In our example, the initial call to the MIXTRAN macro specified a modeltype of AMOUNT, so the same modeltype should be used here. The syntax is:

modeltype=amount,.

pred **Mandatory.** The name of the data set containing predicted values for each subject. This data set was created by a previous execution of the MIXTRAN macro. The input data to be used depends on the modeltype. If the modeltype in the MIXTRAN macro was CORR and the correlated model was fit successfully, then the correlated data set can be input to the DISTRIB macro. If the modeltype in the MIXTRAN macro was NOCORR or AMOUNT, then only the data set labeled with "_unc" can be input to the DISTRIB macro. The full name of the data set is also determined by the value of the parameters "outlib", "foodtype" and "vcontrol" that were supplied in the call to the MIXTRAN macro. Please see the section SAS Data Sets Saved by the MIXTRAN Macro for an explanation of data set naming conventions. In our example, the library reference is specified as "outlib=mylib"; the "modeltype" parameter is "amount"; the "foodtype" parameter is "add_sug" and the "vcontrol" parameter is a null string. The syntax is:

pred=mylib._pred_unc_add_sug,.

param **Mandatory.** The name of the data set containing the parameter values. This data set was created by a previous execution of the MIXTRAN macro. The input data to be used depends on the modeltype. If the modeltype in the MIXTRAN macro was CORR and the correlated model was fit successfully, then the correlated data set can be input to the DISTRIB macro. If the modeltype in the MIXTRAN macro was NOCORR or AMOUNT, then only the data set labeled with "_unc" can be input to the DISTRIB macro. The full name of the data set is determined also by the value of the parameters "outlib", "foodtype" and "vcontrol" that were supplied in the call to the MIXTRAN macro. Please see the section SAS Data Sets Saved by the MIXTRAN Macro for an explanation of data set naming conventions. In our example, the library reference is specified as "outlib=mylib"; the "modeltype" parameter is "amount"; the "foodtype" parameter is "add_sug" and the "vcontrol" parameter is a null string. The syntax is:

param=mylib._param_unc_add_sug,.

outlib **Mandatory.** The library reference to the parameter and predicted data sets saved by the MIXTRAN macro, and the directory to which the data set of distributions will be written by the DISTRIB macro. Since the input data sets for the DISTRIB macro were output by the MIXTRAN macro, the value of "outlib" should be identical to the value of the "outlib" parameter in the MIXTRAN macro. An explanation of the naming conventions for the data set

output by the DISTRIB macro follows the explanation of the parameters. In our example, the library name is "mylib", so the syntax is:

outlib=mylib,.

cutpoints **Optional.** A list of the cutoff points, each separated by a single space. If no cutoff points are supplied, then the DISTRIB macro will only calculate the mean and percentiles of intake. Our example includes cutoff points, and the syntax is:

cutpoints= 10 12 14 17 22 25 33 38 42 47,.

ncutpnt **Optional.*** The number of cutoff points in the cutpoint list. This must be an integer. * The ncutpnt parameter is **mandatory** if the parameter cutpoints is used. Since we have 10 cutpoints in our example, the syntax is:

ncutpnt=10,.

byvar **Optional.** A list of by-variables that are in the parameter and predicted data sets, indicating that the MixTran model was fit separately for each by group. That is, a separate call to the MIXTRAN macro was made for each by group. (For example if males and females were each passed through the MIXTRAN macro separately, the parameter and predicted data sets for each could be concatenated, merging the appropriate parameter estimates to the predicted values, and passed to the DISTRIB macro together.) The DISTRIB macro will produce distributions for the entire population, not distributions within each bygroup. Use the "subgroup" parameter to obtain distributions for subpopulations. The "byvar" and "subgroup" parameters can be used together. Our example does not use by variables, so the syntax is:

byvar=,.

subgroup **Optional.** A single categorical variable used for the calculation of a separate usual intake distribution for each subgroup. The distribution of usual intake will also be calculated for the overall data set (i.e. all subjects). The subgroup variable must also be in the call to the MIXTRAN macro. Our example includes a subgroup variable named "agegroup", so the syntax is:

subgroup=agegroup,.

subject **Optional.*** A variable that uniquely identifies each subject. This information is only needed if the MIXTRAN macro was a weekend run. * If the MIXTRAN macro was a weekend run then this parameter is **mandatory**. In our example, the call to the MIXTRAN macro did invoke the "weekend" parameter, so the syntax is:

subject=seqn,.

titles **Optional.** An integer from 0 to 4. Up to 4 title lines can be reserved for custom titles in the .lst output. Our example reserves 4 title lines. The syntax is:

titles=4,.

food **Optional.** A one word description of the food being analyzed. The value of this parameter is included in the name of the data set output by the DISTRIB macro, to distinguish it from similar data sets output by other calls to the DISTRIB macro. The string must be valid in a SAS data set name. In our example, this is the last parameter to be entered, and therefore it is not followed by a comma. The syntax is:

food=add_sug.

SAS Data Sets Saved by the DISTRIB Macro

The DISTRIB macro outputs one SAS data set. This data set contains descriptive statistics for the usual intake. These are the mean and percentiles from the distribution of usual intake for the population being analyzed, and optionally the cutpoint probabilities. There is one record for all subjects in the data set, and if a subgroup was declared there is an additional record for each level of the subgroup variable. The data set is named using the following conventions:

outlib.**descript**_food_freq_var

where:

outlib is the library name given in the DISTRIB parameter "outlib" thus depends on user input.

descript is the name used to distinguish the distribution data set and is always "descript".

_food is obtained from the "food" parameter value supplied to the DISTRIB macro thus depends on user input.

_freq_var is obtained from the "replicate_var" parameter value supplied to the MIXTRAN macro thus depends on user input. If no replicate variable was supplied, then this value will be a null string. The name of this variable was saved in the parameter data sets and is therefore available to the DISTRIB macro.

Thus, in our above example of MIXTRAN and DISTRIB macro calls, where the library reference is "mylib", the food is "add_sug" and the replicate_var is "rndw1", the output data set will be named:

mylib.descript_add_sug_rndw1.

Notes to the User

Because of the complexity of the model, the MIXTRAN macro can require considerable computing time. When the model type is **amount** (i.e. **modeltype=amount**), we have seen that the macro generally requires less than 10 minutes of computing time for NHANES data. When the model type is **uncorrelated** (i.e. **modeltype=nocorr**) or **correlated** (i.e. **modeltype=corr**), however, the required computing time can be considerably longer; in analyzing NHANES data, we have seen correlated models require from as little as 20 minutes to 2 hours or longer.

Check the log file for lines beginning with "##". These lines act as informal documentation of the parameters used in the MIXTRAN and DISTRIB macros.

Base runs are defined as being a call to the MIXTRAN macro with no user supplied starting values to the three SAS NLMixed procedures; therefore, start_val1, start_val2, and start_val3 will not be assigned a value. To obtain variance estimates using BRR, the same models are run numerous times with different weights. Instead of letting the MIXTRAN macro calculate the starting values for each call to the SAS NLMixed procedure, the parameter estimates output by the SAS NLMixed procedure from the first run (i.e. base run) can be used as starting values.

It is possible to re-run the DISTRIB macro without re-running the MIXTRAN macro once the MIXTRAN macro has successfully executed. The data sets needed for input to the DISTRIB macro have been saved and are available for use at any time. For example, if the user decides to try different cutoffs (i.e. cutpoints) in the DISTRIB macro, it would not be necessary to re-run the MIXTRAN macro. However, please be absolutely sure to use the proper data sets as input to the DISTRIB macro. Example 3, discussed in the next section, provides another example of using the DISTRIB macro without re-running the MIXTRAN macro.

At the end of the DISTRIB macro execution, the data sets in the work library are deleted. To keep the data set _mcsim1 to use for further analysis in the program, look for the phrase "to keep _mcsim1 for further analysis" in the DISTRIB macro code. This search will find the correct line, which can then be commented out. Since _mcsim1 contains all the simulated records, it might use a lot of storage space.

Examples

The website includes 3 SAS example programs and output files that illustrate the use of the MIXTRAN and DISTRIB macros. The following description of the example programs considers two 24-hour recall variables, "add_sug" and "v_potato", for added sugar and white potato cup equivalents, respectively. The covariates considered include: sequence, weekend, age group, race, and sex. The sequence covariate equals 0 for the first day of recall and equals 1 for the second day of recall, and the weekend covariate equals 1 for Friday-Sunday and equals 0 otherwise. A stratification variable, named "stra", is used to designate the following 3 strata: children age 1-8, males age 9+, and females age 9+. A weight variable, named "rndw1", is also utilized.

Example 1

Example 1 demonstrates use of the MIXTRAN and DISTRIB macros for a food consumed nearly every day. The data set includes males with an age of 9 or older. The model type is "amount" and the food of interest is added sugar. The covariates are sequence, weekend, age group, and race. The sequence covariate and the weekend covariate are entered into the macro using the "seq" and "weekend" parameters, respectively. The "subgroup" parameter is assigned the value "agegroup" and the "replicate_var" parameter is assigned "rndw1" which is the name of the weight variable. An option is added to the NLMixed procedure call. The parameter and predicted data sets output by MIXTRAN for use in the DISTRIB macro are named "mylib._param_unc_add_sug" and "mylib._pred_unc_add_sug".

The DISTRIB macro calculates the percentiles, cutpoint probabilities, and mean intake, by subgroup level and for all levels combined, and writes out a data set named "mylib.descript_add_sug_rndw1". The name of the weight variable used in the MIXTRAN macro is passed to the DISTRIB macro through a saved data set and is used in naming the output data file. The same subgroup variable that was named in the call to the MIXTRAN macro is used in the DISTRIB macro. Distributions are calculated for each subgroup level and for all levels combined. If the subgroup variable is a character variable, then "_overall" is used as a label for the results from all levels combined. In this example, the variable "agegroup" is not a character variable, so the overall group is assigned an "agegroup" value of -255.

Example 2

Example 2 demonstrates use of the MIXTRAN and DISTRIB macros for a food consumed episodically. The data set includes children with an age of 1 to 8. The model type is "corr" (i.e. correlated) and the food of interest is white potato cup equivalents. The covariates are sequence, weekend, age group, race, and sex. The sequence covariate and the weekend covariate are entered into the macro using the "seq" and "weekend" parameters, respectively. The "subgroup" parameter is assigned the value "agegroup" and the "replicate_var" parameter is assigned "rndw1" which is the name of the weight variable. An option is added to the NLMixed procedure call. The parameter and predicted data sets output by MIXTRAN for use in the DISTRIB macro are named "mylib._param_v_potato" and "mylib._pred_v_potato".

The DISTRIB macro calculates the percentiles, cutpoint probabilities, and mean intake, by subgroup level and for all levels combined, and writes out a data set named "mylib.descript_v_potato_rndw1". The name of the weight variable used in the MIXTRAN macro is passed to the DISTRIB macro through a saved data set and is used in naming the output data file. The same subgroup variable that was named in the call to MIXTRAN is used in DISTRIB. Distributions are calculated for each subgroup level and for all levels combined. If the subgroup variable is a character variable, then "_overall" is used as a label for the results from all levels combined. In this example, the variable "agegroup" is not a character variable, so the overall group is assigned an "agegroup" value of -255.

Example 3

Example 3 shows two ways to minimize the time and effort used to produce the distributions of usual intake. The first section minimizes the number of calls to the DISTRIB macro for strata, and the second section

demonstrates a way to re-use the data output by the MIXTRAN macro in follow-up calls to the DISTRIB macro. This approach requires only a little SAS coding outside of the macros.

Three strata of the data (children age 1-8, males age 9+, and females age 9+) are each run through the MIXTRAN macro separately to get independent estimates. The parameters are nearly the same as those used in Example 1. Notice that the subgroup variable is carefully coded so there are no overlapping values between strata. This fact will be important later. The "foodtype" parameter in the call to the MIXTRAN macro is used to create output data files with distinct names. In this example the values of "foodtype" are "add_sug_child", "add_sug_male" and "add_sug_female". The MIXTRAN macro created the data files: "mylib._param_unc_add_sug_child"; "mylib._pred_unc_add_sug_child"; "mylib._param_unc_add_sug_male"; "mylib._pred_unc_add_sug_male"; "mylib._param_unc_add_sug_female"; and "mylib._pred_unc_add_sug_female".

In the SAS program, after the successful execution of all three calls to the MIXTRAN macro but prior to calling the DISTRIB macro, the parameter files for the three strata are concatenated, and the predicted data files are also concatenated. The variable "stra" is coded in both the concatenated parameter file and the concatenated predicted file to designate the appropriate stratum for each record.

The DISTRIB macro is then called, in a manner very similar to the call in Example 1, but this time the "byvar" parameter is invoked for the variable "stra" (i.e. "byvar=stra"). The parameter and predicted data sets will be merged by the variable "stra" thus ensuring that the appropriate parameter estimates are attached to each record in the predicted data set.

The DISTRIB macro will calculate the count, percentiles, cutpoint probabilities, and mean for each level of the subgroup, and for all subjects combined, and save the data in one descriptive file. In this case the name would be "mylib.descript_add_sug_all_rndw1".

The next section of Example 3 demonstrates an instance of avoiding unnecessary calls to the MIXTRAN macro, by using the saved parameter and predicted data sets in new calls to the DISTRIB macro.

In the SAS program the data sets saved by MIXTRAN for the male stratum are subset to males age 19 or older. The variable "stra" is created and assigned a value of 2 in both the parameter and predicted data sets for this subset.

The DISTRIB macro is then called. The subgroup option is omitted because only the combined distribution for all males age 19 or older is required. However, the "byvar" parameter MUST be invoked with the variable "stra" which is the stratification variable. The reason for this requirement is that the weights and subject counts have to be recalculated for the subset of the population. The value of the "food" parameter is changed so that the previous descriptive data set is not overwritten. In the example, the data set output by this execution of DISTRIB is called "mylib.descript_add_sug_m19_rndw1". The data contains the counts, mean, percentiles, and cutpoint probabilities for intake of added sugar by males age 19 and older. Note, that the descriptive data set will not include a subgroup variable. If the descriptive data set is later concatenated with the first descriptive data set produced using subgroup as above, the data from this file will need to be assigned a distinct value for the subgroup variable.

Similarly the data saved by the MIXTRAN macro for the female stratum is subset to females age 19 or older. The variable "stra" is created and assigned a value of 3.

The DISTRIB macro is called, with no subgroup parameter, and the byvar set to "byvar=stra". The food parameter is changed, and another data set of descriptive statistics is saved. It will contain the count, mean, percentiles, and cutpoint probabilities for intake of added sugar by females age 19 or older.

WARNING: SAS VERSION 9 BUG

In SAS version 9 there is a bug in the SAS procedure NLMixed.
If the effect names in the model become too long and too numerous, the procedure will never complete. It has not been possible to discover a definitive limit to the size of the model. This problem has not been encountered in SAS version 8.

According to SAS technical support as of 5/2/2007

"..there is no definitive answer to the size of the model (in terms of pure symbol or character storage) that PROC NLMixed can handle. NLMixed has to store derivatives of the model behind the scenes and those derivatives can double or even quadruple the size of the "symbol space" needed to process the model. It is best to stick with shorter effect names when creating a model in NLMixed, at least until SAS 9.2 comes out. Tech Support does not have an official release date for that version of SAS yet."