

# ***IVEware*: Imputation and Variance Estimation Software**

## **User Guide**

**T. E. Raghunathan**

(teraghu@isr.umich.edu)

**Peter W. Solenberger**

(pws@isr.umich.edu)

**John Van Hoewyk**

(johnvh@isr.umich)



Survey Methodology Program  
Survey Research Center, Institute for Social Research  
University of Michigan

March 2002

***IVEware: Imputation and Variance Estimation Software***

---

COPYRIGHT © 2002  
THE REGENTS OF THE UNIVERSITY OF MICHIGAN  
ALL RIGHTS RESERVED

PERMISSION IS GRANTED TO USE, COPY AND REDISTRIBUTE THIS SOFTWARE FOR ANY PURPOSE, SO LONG AS NO FEE IS CHARGED AND SO LONG AS THE COPYRIGHT NOTICE ABOVE, THIS GRANT OF PERMISSION, AND THE DISCLAIMER BELOW APPEAR IN ALL COPIES MADE; AND SO LONG AS THE NAME OF THE UNIVERSITY OF MICHIGAN IS NOT USED IN ANY ADVERTISING OR PUBLICITY PERTAINING TO THE USE OR DISTRIBUTION OF THIS SOFTWARE WITHOUT SPECIFIC, WRITTEN PRIOR AUTHORIZATION. PERMISSION TO MODIFY OR OTHERWISE CREATE DERIVATIVE WORKS OF THIS SOFTWARE IS NOT GRANTED.

THIS SOFTWARE IS PROVIDED AS IS, WITHOUT REPRESENTATION AS TO ITS FITNESS FOR ANY PURPOSE, AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE REGENTS OF THE UNIVERSITY OF MICHIGAN SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WITH RESPECT TO ANY CLAIM ARISING OUT OF OR IN CONNECTION WITH THE USE OF THE SOFTWARE, EVEN IF IT HAS BEEN OR IS HEREAFTER ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

**Table of Contents**

<b>1. Introduction</b>	5
<b>2. Executing <i>IVEware</i> Module</b>	7
2.1. Execution Mode	7
2.1.1. Interactive Mode	7
2.1.2. Batch Mode	9
<b>3. The IMPUTE Module</b>	11
3.1. IMPUTE Statements	12
3.2. IMPUTE Setup File	19
3.3. IMPUTE List File	19
3.4. Retrieving Multiple Imputed Data Sets	30
<b>4. The DESCRIBE Module</b>	31
4.1. DESCRIBE Statements	31
4.2. DESCRIBE Setup File	36
4.3. DESCRIBE List File	36
4.4. DESCRIBE/IMPUTE Combined	42
4.4.1. Better DESCRIBE Analysis with Incomplete Data	44
<b>5. The REGRESS Module</b>	45
5.1. REGRESS Statements	45
5.2. REGRESS Setup File	51
5.3. REGRESS List File	52
5.3.1. Estimates Statement Output	56
5.3.2. Plot Statement Output	57
5.4. REGRESS/IMPUTE Combined	60
5.4.1. Better REGRESS Analysis with Incomplete Data	61
<b>6. The SASMOD Module</b>	65
6.1. SASMOD Statements	65
6.2. SASMOD Setup File	68
6.3. SASMOD List File	69
<b>References</b>	72
<b>Appendix</b>	73



## 1. Introduction

*IVEware* is a SAS callable software application that can:

1. Perform single or multiple imputations of missing values using the Sequential Regression Imputation Method described in the article “A multivariate technique for multiply imputing missing values using a sequence of regression models” by Raghunathan, Lepkowski, Van Hoewyk and Solenberger (*Survey Methodology*, June 2001).
2. Perform a variety of descriptive and model based analyses accounting for such complex design features as clustering, stratification and weighting.
3. Perform multiple imputation analyses for both descriptive and model-based survey statistics.

*IVEware* is built on the SAS Macro Language and a set of independent C and FORTRAN routines. Users of *IVEware* should have a moderate amount of SAS experience including familiarity with basic file concepts, naming conventions and command file structures. Knowledge of SAS Macro Language, C or FORTRAN is not required.

*IVEware* is currently available for personal computers using the Microsoft Windows and Linux operating systems, and UNIX workstations using Sun Solaris, IBM AIX, and Compaq/DEC Alpha Tru64Unix operating systems. *IVEware* requires SAS version 6.12 or higher.

*IVEware* includes four modules: **IMPUTE**, **DESCRIBE**, **REGRESS** and **SASMOD**.

- **IMPUTE** uses a multivariate sequential regression approach to imputing item missing values. **IMPUTE** can create multiply imputed data sets.
- **DESCRIBE** estimates the population means, proportions, subgroup differences, contrasts and linear combinations of means and proportions. A Taylor Series approach is used to obtain variance estimates appropriate for a user specified complex sample design. A multiple imputation analysis can be performed when there are missing values.
- **REGRESS** fits linear, logistic, polytomous, Poisson, Tobit and proportional hazard regression models for data resulting from a complex sample design. The repeated replication approach is used to estimate the sampling variances. A multiple imputation analysis can be performed when there are missing values.
- **SASMOD** allows users to take into account complex sample design features when analyzing data with several SAS procedures. Currently the following SAS PROCs can be called: CALIS, CATMOD, GENMOD, LIFEREG, MIXED, NLIN, PHREG, and PROBIT. A multiple imputation analysis can be performed when there are missing values.

**NOTE: To use the SASMOD procedure the SAS data file can only include model variables, “by” variables and design variables (weight, stratum and cluster). No other variables are permitted.**

*IVEware* was developed by the Survey Methodology Program at the University of Michigan’s Survey Research Center, Institute for Social Research and is available to users without cost by download from <http://www.isr.umich.edu/src/smp/ive/>. Available for download are the *IVEware* Installation Guide, application files, User Guide, and example setup files with associated SAS data sets.

This is a beta version of *IVEware*. Please report any problems or send along any comments via e-mail to T. E. Raghunathan at [teraghu@umich.edu](mailto:teraghu@umich.edu), Peter Solenberger at [pws@isr.umich.edu](mailto:pws@isr.umich.edu) or John Van Hoewyk at [johnvh@isr.umich.edu](mailto:johnvh@isr.umich.edu).

## 2. Executing IVEware Modules

All *IVEware* modules require two files. An existing **SAS data file** that you wish to impute or analyze and a **setup file** that provides instructions for your imputation or analysis. **The setup file must have the extension “set.”** For illustration purposes, **mydata.sd2** and **mysetup.set** will be used as the names of the SAS data and *IVEware* setup files, respectively, throughout this document. Similarly, we will use the following naming conventions:

- **myindir** : The name of the directory where the SAS input data set **mydata.sd2** is stored.
- **myoutdir**: The name of the directory where output from *IVEware* will be stored.
- **mylibn**: This will be the SAS libname assignments with *n* denoting a number. For example, the following two statements illustrate the naming conventions used in this manual:

```
libname mylib1 'c:\myindir';
libname mylib2 'c:\myoutdir';
```

**mylib1** is assigned to the input directory **c:\myindir** and **mylib2** is assigned to the output directory **c:\myoutdir**.

### 2.1 Execution Mode

*IVEware* modules can be executed either interactively or in batch mode.

#### 2.1.1. Interactive Mode

In the interactive mode, you can type all the relevant setup commands in the SAS PROGRAM EDITOR window and run it by double clicking the “run” icon on SAS toolbar panel. *IVEware* will create and store the setup file. ***You cannot use the ENHANCED EDITOR in SAS Version 8 when using the interactive mode.***

To execute the **DESCRIBE** module interactively you might submit the following commands from the SAS program editor.

```
LIBNAME MYLIB1 'C:\MYINDIR';
%DESCRIBE(NAME=MYSETUP, DIR=C:\MYOUTDIR, SETUP=NEW);
```

```
DATAIN      MYLIB1.MYDATA;
WEIGHT      FNLWGT2;
STRATUM     STRATUM;
CLUSTER     PSU;
MODEL       MULT;
```

```
MEAN      AGE84 HEIGHT WEIGHT;  
CONTRAST  SEX;  
RUN;
```

In this example,

The name **MYLIB1** is assigned to the directory **C: \MYINDIR** with the SAS libname statement.

The **%DESCRIBE** macro statement invokes the DESCRIBE module.

The three keywords within the parentheses reference the setup file that begins with the keyword **DATAIN**.

**Name** = assigns the file name **mysetup.set** to the setup file.

**Dir** = indicates that the setup file and output files are to be stored in **c:\myoutdir**.

**Setup=New** indicates that this is a new setup file. If a file named MYSETUP already exists in directory C:\MYOUTDIR then it will be replaced by the new file.

DATAIN, WEIGHT, STRATUM, CLUSTER, MODEL, MEAN and CONTRAST are some of the **DESCRIBE** keywords. A complete list of and descriptions for DESCRIBE keywords are provided in section 4.

If you wish to run the same setup later, you need only submit the following statements.

```
LIBNAME MYLIB1 ' C: \MYINDIR' ;  
%DESCRIBE(NAME=MYSETUP, DIR=C: \MYOUTDIR) ;
```

The other *IVEware* modules can be executed the same way.

The **IMPUTE** module statements are,

```
LIBNAME MYLIB1 ' C: \MYINDIR' ;  
%IMPUTE (SETUP=NEW, NAME=MYSETUP, DIR=C: \MYOUTDIR) ;  
Impute key words
```

The **REGRESS** module can be invoked by submitting,

```
LIBNAME MYLIB1 ' C: \MYINDIR' ;  
%REGRESS(SETUP=NEW, NAME=MYSETUP, DIR=C: \MYOUTDIR) ;  
Regress key words
```



The **SASMOD** module statement are,

```
LIBNAME MYLIB1 'C:\MYINDIR';  
%SASMOD(SETUP=NEW, NAME=MYSETUP, DIR=C:\MYOUTDIR);  
Sasmod keywords
```

If *IVEware* has been installed properly and the syntax of your setup file is correct, then submitting your setup file will produce several windows reporting on the module's execution. (At times the windows open and close very quickly.) At the successful completion of the module the results of your imputation or analysis will appear in the SAS output window. The same results are stored in a list file located in the directory that contains your setup file. List files will have the same name as the submitted setup file but have a .lst extension instead of a .set extension. For example, the DESCRIBE setup file illustrated above will create **mysetup.lst** in **c:\myoutdir**.

### 2.1.2. Batch Mode

The batch mode may be useful if you are using a UNIX environment through remote login or processing a large job. In the batch mode you need to create and store two files, an *IVEware* setup file and a SAS command file to read and execute the setup file.

If, for example, you wish to execute the **DESCRIBE** module, you would first create, using a text editor or word processing application, the setup file **mysetup.set** and store it as a text file in **c:\myoutdir**. The file **mysetup.set** may include among others the following *IVEware* commands:

```
DATA IN      MYLIB1.MYDATA;  
WEIGHT      FNLWGT2;  
STRATUM     STRATUM;  
CLUSTER     PSU;  
MODEL       MULT;  
MEAN        AGE84 HEIGHT WEIGHT;  
CONTRAST    SEX;  
RUN;
```

Then you would create and store the SAS command file **myfile.sas**, again using a text editor or a word processing application. The **myfile.sas** file would include:

```
LIBNAME MYLIB1 'C:\MYINDIR';  
%DESCRIBE(NAME=MYSETUP, DIR=C:\MYOUTDIR);
```

At the prompt type:

```
sas myfile <Enter>
```

or to run the job in background

```
sas myfile & <Enter>
```

If the run is successful, you should see **mysetup.lst** in **myoutdir** containing the *IVEware* output.

Example setup files and SAS data sets are included with *IVEware* and can be found in the **ive\_examples.zip** or **ive\_examples.tgz** files. The example setups can be used to test that *IVEware* has been installed properly. The setups and their output are used through out this manual to illustrate the *IVEware* modules.

The SAS data files associated with the example setup files are in SAS export format and can be converted to SAS data files for Windows or Unix by using the PROCCOPY.SAS command file also found in example files.

### 3. IMPUTE

The IMPUTE module is a general-purpose multivariate imputation procedure that can handle relatively complex data structures when the data are missing at random (Rubin, 1976). Survey data sets often consist of large numbers of variables that have a variety of distributional forms. Typically, such data sets have hundreds of variables, some continuous, others counts, many dichotomous or polytomous, and semi-continuous or limited dependent variables. IMPUTE can handle such complex data structures.

IMPUTE produces imputed values for each individual in the data set conditional on all the values observed for that individual. The approach is to consider imputation on a variable-by-variable basis but to condition on all observed variables. The basic strategy is to create imputations through a sequence of multiple regressions, varying the type of regression model by the type of variable being imputed. Covariates include all other variables observed or imputed for that individual. The imputations are defined as draws from the posterior predictive distribution specified by the regression model with a flat or non-informative prior distribution for the parameters in the regression model. The sequence of imputing missing values can be continued in a cyclical manner, each time overwriting previously drawn values, building interdependence among imputed values and exploiting the correlational structure among covariates. To generate multiple imputations, the same procedure can be applied with different random starting seeds or taking every  $p^{th}$  imputed set of values in the cycles mentioned above. For details see Raghunathan et. al. (2001).

IMPUTE assumes the variables in the data set are one of the following five types: (1) continuous; (2) binary; (3) categorical (polytomous with more than two categories); (4) counts; and (5) mixed (a continuous variable with a non-zero probability mass at zero). The types of regression models used are linear, logistic, Poisson, generalized logit or mixed logistic/linear, depending on the type of variable being imputed.

IMPUTE can also accommodate two common features of survey data that add to the complexity of the modeling process: the restriction of imputations to subpopulations, and the bounding of imputed values.

First, certain restrictions are imperative, requiring the subsetting of sample individuals to satisfy particular criteria while fitting the regression models. For example, the variable "Number of Years Since Quit Smoking" is defined only for former smokers; hence, the imputation process for this variable should be restricted only to former smokers. Restrictions also arise due to skip patterns in the questionnaire. For example, certain questions about income from a second job are asked only when the respondent indicates having a second job. The imputation of such variables has to be handled in a hierarchical manner.

Second, there are certain logical or consistency bounds for missing values that must be incorporated

in the imputation process. Such interrelationships among the variables make the model specification difficult. For instance, "Years of Smoking" should not only be restricted to current or past smokers but the imputed values might be required to be less than a specified number years, based on other respondent characteristics, such as evidence of smoking as a teen-ager. In such a case, the imputed upper bound for "Year of Smoking" might be the respondent's current age minus 12. This assumes that the respondent may have started smoking at 12 years of age. For a former smoker, "Year of Smoking" would also have take into account years since the respondent stopped smoking. Another example of bounds is discussed in Heeringa, Little and Raghunathan (1997). They address imputation of bracketed response questions in which a respondent is unable or unwilling to provide an exact response (e.g., income and assets), but does define the bounds within which the imputed values must lie. The bounds involve drawing values from a truncated predictive distribution.

### 3.1. IMPUTE Statements

#### Required or Standard Statements

##### **DATAIN libname.filename;**

This required statement identifies the location and name of the input SAS data set.

For example,

```
DATAIN Mylib1.Mydata;
```

indicates that the SAS data file **Mydata** is located in the library **Mylib1**. **Mylib1** is the name assigned to a directory with the SAS Libname statement. (See section 2 for examples of SAS Libname statements or consult the SAS user manual for a more extensive discussion of Libname.)

##### **DATAOUT libname.outfile [ALL];**

This statement identifies the location and name of the output SAS dataset containing the imputed data. The ALL keyword is optional. If it is specified and more than one imputation is generated (see keyword MULTIPLES) then the output dataset will be a concatenation of the multiple imputed data sets. The system variable `_MULT_`, automatically added to the out put file, can be used to distinguish each imputation.

For example,

```
DATAOUT Mylib2.Impdata ALL;
```

will store the SAS file **Impdata** in the library **Mylib2**.

**DECLARING VARIABLE TYPES**

IMPUTE requires that the SAS data set variables be defined by type. Six types of variables are recognized by the IMPUTE module: continuous, categorical, count, mixed, transfer and drop. If no variable types are specified, all variables will be assumed to be continuous. Variable types should be declared before any BOUNDS, INTERACT, or RESTRICT statements (see below).

**CONTINUOUS variable list;**

Variables declared as CONTINUOUS may take on any value on a continuum. Income is an example of a continuous variable. A normal linear regression model is used to impute the missing values in these variables. You may want to transform the variable to achieve normality and then impute on the transformed scale. After imputation you may re-transform the variable back to its original form.

**CATEGORICAL variable list;**

CATEGORICAL variables have values that represent discrete values. Gender is a categorical variable. A logistic or generalized logistic model is used to impute missing categorical values.

**MIXED variable list;**

Variables declared as MIXED are both categorical and continuous. In a mixed variable a value of zero is treated as a discrete category, while values greater than zero are considered continuous. Alcohol consumption is an example of a mixed variable. A two stage model is used to impute the missing values. First, a logistic regression model is used to impute zero vs. non-zero status. Conditional on imputing a non-zero status, a normal linear regression model is used to impute non-zero values.

**COUNT variable list;**

COUNT variables have non-negative integer values. A Poisson regression model is used to impute the missing values. The number of annual doctor visits is an example of a COUNT variable

**DROP variable list;**

Variables listed after the DROP keyword will be excluded from the imputation procedure and will not appear in the imputed data set.

**TRANSFER variable list;**

Variables listed after the TRANSFER keyword are carried over to the imputed data set, but are not imputed nor used as predictors in the imputation model. Transfer variables, however, can be used in the RESTRICT and BOUNDS statements (see below). ID is an example of a variable that you might want to treat as a transfer

variable.

**DEFAULT variable type;**

**Variable type** can be Continuous, Categorical, Count, Mixed, Transfer or Drop. This keyword declares that by default all the variables in the data set should be treated as the **variable type**. The most efficient use of the DEFAULT statement is to declare the most numerous variable type in your data set as the default type, eliminating the need to type a long list of variables.

**RUN;**

This should be the last statement in your setup file.

**Optional Statements****RESTRICT variable(logical expression);**

This command is used to restrict the imputation of a variable to those observations that satisfy the logical expression. For instance, suppose that the variable **Yrssmoke** indicates the number of years an individual smoked, and the variable **Smoke** takes the value 1 for a current smoker, 2 for a former smoker or 0 for someone who never smoked.

Then the declaration,

```
RESTRICT Yrssmoke(Smoke=1, 2);
```

will impute **Yrssmoke** values only for current and former smokers. It will automatically set **Yrssmoke** equal to 0 for never smokers.

Restrictions on more than one variable may be combined as follows:

```
RESTRICT Yrssmoke(Smoke=1, 2) Births(Gender=2) Income(Employed=1);
```

When the restriction is not met, the value of the restricted variable will be set to zero for continuous variables and one higher than the highest observed code for categorical variables.

**BOUNDS variable (logical expression);**

This keyword is useful for restricting the range of values to be imputed for a variable.

For example,

```
BOUNDS Yrssmoke (> 0, <= Age- 12);
```

will ensure that the imputed values for **Yrssmoke** are between 0 and the individual's Age minus 12. Smoking is assumed not to begin before the age of 12.

Again, as in the RESTRICT statement more than one variable can be included in the BOUNDS statement.

For example,

**BOUNDS Yrssmoke (>0, <= Age- 12) Numci g(>0);**

### **Model-Building Statements**

The following commands are useful in the specification of the imputation model.

#### **INTERACT variable\*variable;**

This keyword enables the users to specify interaction terms to be include in the imputation regression model.

**INTERACT Income\*Income, Age\*Race;**

In this example, the imputation model for all the variables will include a square term for Income and an interaction term of Age and Race.

### **STEPWISE REGRESSION**

#### **MAXPRED number; OR MAXPRED varlist2 (number) ;**

Specifies the maximum number of predictor variables to be included as predictors in the regression model. A step-wise regression procedure is used to select the best predictors subject to the maximum number. Setting MAXPRED to a small number of predictors will greatly reduce the computational time especially for a very large data sets but the imputations will not be fully conditional.

For example,

**MAXPRED 5;**

will include the five best predictor variables, the five making the largest contribution to the r-square.

You can also restrict the number of predictors for selected variables.

**MAXPRED Income (7) Educ (3);**

will limit the number of predictors of Income to the seven largest contributors to the r-square, while the number of predictors of Educ are limited to the three largest

contributors. For other variables, all variables will be used as predictors.

**MINRSQD decimal;**

Specifies the minimum marginal r-squared for a stepwise regression. (Minimum initial marginal r-squared for a logistic regression, and minimum initial r-squares for any code being predicted for a polytomous regression.) This can reduce computation time. A small decimal number like 0.005 would build very large regression models whereas 0.25 will include a smaller number of predictors in the regression models. If neither MAXPRED nor MINRSQD is set then no stepwise regression will be preformed.

**MINRSQD 0.01;**

In this example, only variables with minimum additional r-square of 0.01 or higher will be included as predictors.

**MAXLOGI number;**

Specifies the maximum number of iterative algorithms to be performed in a logistic or multilogit regression model. The default is 50. This is useful if the Newton-Raphson algorithm used in producing maximum likelihood estimates does not converge after 50 iterations. This applies to the convergence criterion for the logistic, polytomous and Poisson regression models. You can check whether you have such a non-convergence problem by inspecting the log file (e.g., mysetup.log).

**MINCODI decimal;**

Specifies the minimum proportional change in any regression coefficient to continue the logistic regression iteration process. This applies to the convergence criterion for the logistic, polytomous and Poisson regression models.

**ITERATIONS number;**

Specifies the number of cycles you would like the imputation program to carry out. You can specify any number greater than or equal to 2. Current investigations show that about 10 cycles are sufficient for most imputations. You may want to experiment with several values and check the differences in the resulting analysis.

**MULTIPLES number;**

Indicates the number of imputations to be performed. By default only a single imputation is generated. Multiples and iterations determine  $p$  (see page 11). If multiples were specified as 5 and iterations as 10 then a total of 50 cycles will be performed. After every 10<sup>th</sup> cycle an imputed data set will be created.



(See section 3.4 for more information about multiple imputations.)

**PERTURB instruction;**

The keyword PERTURB followed by an instruction (COEF/SIR) allows the user to control perturbations of imputed values. By default the IMPUTE module will perturb model coefficients using a multivariate normal approximation of the posterior distribution and the predicted values using the appropriate regression model conditional on the perturbed coefficients. This is equivalent to using the COEF instruction. SIR uses the Sampling-Importance-Resampling algorithm to generate coefficients from the actual posterior distribution of parameters in the logistic, polytomous and Poisson regression models (See Rubin 1987a, Raghunathan and Rubin 1988, Raghunathan 1994, Gelman, et. al 1995). This is appropriate in situations where normal approximation to the posterior distribution is not appropriate.

**PERTURB Si r;**

**SEED number;**

Specifies a seed for the random draws from the posterior predictive distribution.

**Number** should be greater than zero. A zero seed will result in no perturbations of the predicted values or the regression coefficients. If the SEED keyword is missing from the setup file then the seed will be determined by your computer's internal clock.

**NOBS number;**

NOBS indicates the number of observations to be used in the analysis. By default all observations in the data set will be used. You might use NOBS to subset a large data set while testing your setup file.

**OFFSETS count variables (offset variable) ;**

This statement is used to specify an offsets variable when fitting a Poisson regression model.

For example,

**OFFSETS Injuri es(Years) ;**

will fit a model predicting the number for injuries occurring per year.

**PRINT instruction;**

Indicates the printout desired. The options are STANDARD, DETAILS, COEF, and ALL.

For the IMPUTE procedure, the STANDARD and DETAILS keywords instruct *IVEware*

to print the number and distribution of observed values, imputed values, and combined observed and imputed values for each variable.

The keyword COEF instructs IMPUTE to also print the unperturbed and perturbed coefficients for each iteration of each multiple imputation. When the ALL keyword is used, in addition to the above, the coefficient covariance matrix for each iteration of each multiple imputation is also printed.

IMPUTE also printouts a list of the variables used in the imputation model with columns indicating the number of observed cases and the number of imputed cases for each of the variables.

The third column of the variable list, labeled “double counted,” is to be used for diagnostic purposes. This entry should be zero. A non-zero entry indicates that the imputed value of a restricting variable has caused the observed value of a restricted variable to be set to the restricted value (zero for continuous variables, one higher than the highest observed code for categorical variables; see RESTRICT above). This usually indicates a mis-specification of the restriction or an inconsistency in the observed data. In either case, you need to run a data step before the imputation to check the appropriateness of the restriction or correct the data inconsistency.

For example, if the variable SMOKE, indicating whether or not a respondent smokes, is missing and the variable YRSMK, indicating the number of years the respondent has smoked, is observed, then logically the respondent should be classified as a smoker. If SMOKE is not given a value indicating the respondent is a smoker in a SAS data step prior to imputation, the missing value could possibly be imputed to a nonsmoker value, causing IMPUTE to change the observed value for YRSMK to zero.

**TITLE text \n text;**

Indicates the title(s) to be printed at the top of each page of the printout. A \n indicates that the text that follows should be printed on the next line.

For example,

**TITLE This is the title on the first line \n This is the title on the second line;**

**3.2. IMPUTE Setup File**

The following is an example of an IMPUTE setup file named MYSETUP.<sup>1</sup>

```
LIBNAME MYLIB 'C:\MYINDIR';
LIBNAME MYOUT 'C:\MYOUTDIR';

%IMPUTE (NAME=MYSETUP, DIR=C:\MYOUTDIR, SETUP=NEW);
  DATAIN      MYLIB.MYDATA2;
  DATAOUT     MYOUT.IMPDATA;
  DEFAULT      CONTINUOUS;
  CATEGORICAL  CASECNT GENDER RACE3 HYPER DIAB
              SMOKE  FAMMI EDUSUBJ3 CHOLESTH;
  MIXED        CAFFTOT ALCOHOL3;
  TRANSFER     STUDYID ;
  RESTRICT     NUMCIG(SMOKE=2, 3) YRSSMOKE(SMOKE=2, 3);
  BOUNDS       NUMCIG(>0)          YRSSMOKE(>0, <=AGE-12)
              FATINDEX (>0)       CAFFTOT (>=0) ALCOHOL3 (>=0);
  MAXPRED      REDTOT(3) WGTKG (2);
  MINRSQD      .01;
  ITERATIONS   5;
  MULTIPLES    2;
  SEED         2001;
RUN;
```

**3.3. IMPUTE List File**

Setup file MYSETUP (see 3.2) produces the following in the SAS output window. The same output is stored in a list file placed in the directory where the setup file is stored (e.g.,myoutdir). List files have the same name as the setup file but have a .lst extension. (In this case, MYSETUP.LST.)

IVEware Setup Checker, Wed Sep 19 13:08:24 2001

1

Setup listing:

```
DATAIN      MYLIB.MYDATA2;
DATAOUT     MYLIB.IMPDATA;
DEFAULT      CONTINUOUS;
CATEGORICAL CASECNT GENDER RACE3 HYPER DIAB
              SMOKE  FAMMI EDUSUBJ3 CHOLESTH;
MIXED        CAFFTOT ALCOHOL3;
```

<sup>1</sup> This setup is included in the example files available at [www.isr.umich.edu/src/smp/ive/](http://www.isr.umich.edu/src/smp/ive/). See IMPUTE.SAS.

```

TRANSFER    STUDYID ;
RESTRICT    NUMCIG(SMOKE=2, 3) YRSSMOKE(SMOKE=2, 3);
BOUNDS      NUMCIG(>0)          YRSSMOKE(>0, <=AGE- 12)
            FATINDEX (>0)      CAFFTOT (>=0) ALCOHOL3 (>=0);
MAXPRED     REDTOT(3) WGTKG (2);
MINRSQD     . 01;
ITERATIONS  5;
MULTIPLES   2;
SEED        2001;
RUN;
    
```

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001 1

**Imputation 1**

Variable	Observed	Imputed	Double counted
CASECNT	898	0	0
AGE	898	0	0
GENDER	898	0	0
RACE3	897	1	0
HYPER	890	8	0
DIAB	894	4	0
SMOKE	896	2	0
NUMCIG	454	444	0
YRSSMOKE	489	409	0
FATINDEX	864	34	0
FAMMI	891	7	0
EDUSUBJ3	898	0	0
DHA_EPA	898	0	0
REDTOT	498	400	0
CHOLESTH	884	14	0
CAFFTOT	896	2	0
WGTKG	845	53	0
TOTLKCAL	898	0	0
ALCOHOL3	897	1	0
HGTCM	896	2	0

**Variable CASECNT**

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent

0	551	61.36	551	61.36
1	347	38.64	347	38.64
Total	898	100.00	898	100.00

Variable AGE

	Observed		Imputed		Combined	
Number		898				898
Minimum		29				29
Maximum		79				79
Mean		58.6303				58.6303
Std Dev		10.1851				10.1851

Variable GENDER

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	705	78.51			705	78.51
1	193	21.49			193	21.49
Total	898	100.00			898	100.00

Variable RACE3

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	55	6.13	0	0.00	55	6.12
1	842	93.87	1	100.00	843	93.88
Total	897	100.00	1	100.00	898	100.00

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001

2

Variable HYPER

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	709	79.66	5	62.50	714	79.51
1	181	20.34	3	37.50	184	20.49
Total	890	100.00	8	100.00	898	100.00

Variable DIAB

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	831	92.95	4	100.00	835	92.98
1	63	7.05	0	0.00	63	7.02

Total	894	100.00	4	100.00	898	100.00
-------	-----	--------	---	--------	-----	--------

Variable SMOKE

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
1	354	39.51	2	100.00	356	39.64
2	364	40.63	0	0.00	364	40.53
3	178	19.87	0	0.00	178	19.82
Total	896	100.00	2	100.00	898	100.00

Variable NUMCIG

	Observed	Imputed	Combined
Number	454	444	898
Minimum	0	0	0
Maximum	98	51.5835	98
Mean	22.7004	4.43452	13.6692
Std Dev	14.3406	10.0891	15.415

Variable YRSSMOKE

	Observed	Imputed	Combined
Number	489	409	898
Minimum	1	0	0
Maximum	63	24.4291	63
Mean	28.1554	0.913425	15.7479
Std Dev	14.677	3.07985	17.4863

Variable FATINDEX

	Observed	Imputed	Combined
Number	864	34	898
Minimum	10	14.0127	10
Maximum	33	33.3627	33.3627
Mean	21.5625	22.9625	21.6155
Std Dev	3.94349	4.00221	3.95252

Variable FAMMI

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	487	54.66	0	0.00	487	54.23
1	404	45.34	7	100.00	411	45.77
Total	891	100.00	7	100.00	898	100.00

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001

3

Variable EDUSUBJ3

Code	Observed		Imputed		Combined	
	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	258	28.73			258	28.73
1	640	71.27			640	71.27
Total	898	100.00			898	100.00

Variable DHA\_EPA

	Observed	Imputed	Combined
Number	898		898
Minimum	0		0
Maximum	42.7166		42.7166
Mean	4.91028		4.91028
Std Dev	5.72728		5.72728

Variable REDTOT

	Observed	Imputed	Combined
Number	498	400	898
Minimum	1.97	1.02914	1.02914
Maximum	10.937	9.0387	10.937
Mean	4.64733	4.57884	4.61682
Std Dev	1.17307	1.19381	1.18218

Variable CHOLESTH

Code	Observed		Imputed		Combined	
	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	682	77.15	10	71.43	692	77.06
1	202	22.85	4	28.57	206	22.94
Total	884	100.00	14	100.00	898	100.00

Variable CAFFTOT

	Observed	Imputed	Combined
Number	896	2	898
Minimum	0	0	0

Maximum	4120.2	448.6	4120.2
Mean	381.696	224.3	381.345
Std Dev	462.207	317.208	461.873

Variable **WGTKG**

	Observed	Imputed	Combined
Number	845	53	898
Minimum	43.5446	39.4438	39.4438
Maximum	147.417	107.355	147.417
Mean	81.3204	80.4025	81.2662
Std Dev	16.3584	13.6671	16.2068

Variable **TOTLKCAL**

	Observed	Imputed	Combined
Number	898		898
Minimum	0		0
Maximum	23557.7		23557.7
Mean	1195.84		1195.84
Std Dev	1587.4		1587.4

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001

4

Variable **ALCOHOL3**

	Observed	Imputed	Combined
Number	897	1	898
Minimum	0	5.518	0
Maximum	18.4	5.518	18.4
Mean	0.915778	5.518	0.920903
Std Dev	1.73589	0	1.74171

Variable **HGTCM**

	Observed	Imputed	Combined
Number	896	2	898
Minimum	142.24	175.027	142.24
Maximum	203.2	178.29	203.2
Mean	176.221	176.658	176.222



Std Dev                    9. 22544                    2. 30744                    9. 2155

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001                    5

**Imputation 2**

Variable	Observed	Imputed	Double counted
CASECNT	898	0	0
AGE	898	0	0
GENDER	898	0	0
RACE3	897	1	0
HYPER	890	8	0
DIAB	894	4	0
SMOKE	896	2	0
NUMCIG	454	444	0
YRSSMOKE	489	409	0
FATINDEX	864	34	0
FAMMI	891	7	0
EDUSUBJ3	898	0	0
DHA_EPA	898	0	0
REDTOT	498	400	0
CHOLESTH	884	14	0
CAFFTOT	896	2	0
WGTKG	845	53	0
TOTLKCAL	898	0	0
ALCOHOL3	897	1	0
HGTCM	896	2	0

**Variable CASECNT**

Code	Observed		Imputed		Combined	
	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	551	61.36			551	61.36
1	347	38.64			347	38.64
Total	898	100.00			898	100.00

**Variable AGE**

	Observed	Imputed	Combined
Number	898		898
Minimum	29		29
Maximum	79		79
Mean	58.6303		58.6303
Std Dev	10.1851		10.1851

Variable GENDER

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	705	78.51			705	78.51
1	193	21.49			193	21.49
Total	898	100.00			898	100.00

Variable RACE3

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	55	6.13	0	0.00	55	6.12
1	842	93.87	1	100.00	843	93.88
Total	897	100.00	1	100.00	898	100.00

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001

6

Variable HYPER

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	709	79.66	5	62.50	714	79.51
1	181	20.34	3	37.50	184	20.49
Total	890	100.00	8	100.00	898	100.00

Variable DIAB

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	831	92.95	4	100.00	835	92.98
1	63	7.05	0	0.00	63	7.02
Total	894	100.00	4	100.00	898	100.00

**Variable SMOKE**

Code	Observed		Imputed		Combined	
	Frequency	Percent	Frequency	Percent	Frequency	Percent
1	354	39.51	1	50.00	355	39.53
2	364	40.63	1	50.00	365	40.65
3	178	19.87	0	0.00	178	19.82
Total	896	100.00	2	100.00	898	100.00

**Variable NUMCIG**

	Observed	Imputed	Combined
Number	454	444	898
Minimum	0	0	0
Maximum	98	61.8713	98
Mean	22.7004	3.28597	13.1013
Std Dev	14.3406	8.48633	15.2889

**Variable YRSSMOKE**

	Observed	Imputed	Combined
Number	489	409	898
Minimum	1	0	0
Maximum	63	60.4744	63
Mean	28.1554	3.41366	16.8866
Std Dev	14.677	10.1157	17.7688

**Variable FATINDEX**

	Observed	Imputed	Combined
Number	864	34	898
Minimum	10	10.4713	10
Maximum	33	35.6304	35.6304
Mean	21.5625	21.2185	21.5495
Std Dev	3.94349	4.86552	3.97956

**Variable FAMMI**

Code	Observed		Imputed		Combined	
	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	487	54.66	5	71.43	492	54.79
1	404	45.34	2	28.57	406	45.21
Total	891	100.00	7	100.00	898	100.00

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001

7

## Variable EDUSUBJ3

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	258	28.73			258	28.73
1	640	71.27			640	71.27
Total	898	100.00			898	100.00

## Variable DHA\_EPA

	Observed	Imputed	Combined
Number	898		898
Minimum	0		0
Maximum	42.7166		42.7166
Mean	4.91028		4.91028
Std Dev	5.72728		5.72728

## Variable REDTOT

	Observed	Imputed	Combined
Number	498	400	898
Minimum	1.97	1.89582	1.89582
Maximum	10.937	8.28802	10.937
Mean	4.64733	4.56212	4.60937
Std Dev	1.17307	1.09653	1.13977

## Variable CHOLESTH

	Observed		Imputed		Combined	
Code	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	682	77.15	9	64.29	691	76.95
1	202	22.85	5	35.71	207	23.05
Total	884	100.00	14	100.00	898	100.00

## Variable CAFFTOT

	Observed	Imputed	Combined
Number	896	2	898
Minimum	0	0	0
Maximum	4120.2	30.6708	4120.2
Mean	381.696	15.3354	380.88
Std Dev	462.207	21.6876	462.015

**Vari able WTKG**

	<b>Observed</b>	<b>Imputed</b>	<b>Combi ned</b>
<b>Number</b>	845	53	898
<b>Mi ni mum</b>	43. 5446	43. 9416	43. 5446
<b>Maxi mum</b>	147. 417	122. 928	147. 417
<b>Mean</b>	81. 3204	83. 0484	81. 4224
<b>Std Dev</b>	16. 3584	17. 5388	16. 4251

**Vari able TOTLKCAL**

	<b>Observed</b>	<b>Imputed</b>	<b>Combi ned</b>
<b>Number</b>	898		898
<b>Mi ni mum</b>	0		0
<b>Maxi mum</b>	23557. 7		23557. 7
<b>Mean</b>	1195. 84		1195. 84
<b>Std Dev</b>	1587. 4		1587. 4

IVEware Iterative Imputation Procedure, Wed Sep 19 13:08:29 2001

8

**Vari able ALCOHOL3**

	<b>Observed</b>	<b>Imputed</b>	<b>Combi ned</b>
<b>Number</b>	897	1	898
<b>Mi ni mum</b>	0	0	0
<b>Maxi mum</b>	18. 4	0	18. 4
<b>Mean</b>	0. 915778	0	0. 914758
<b>Std Dev</b>	1. 73589	0	1. 73519

**Vari able HGTCM**

	<b>Observed</b>	<b>Imputed</b>	<b>Combi ned</b>
<b>Number</b>	896	2	898
<b>Mi ni mum</b>	142. 24	157. 087	142. 24
<b>Maxi mum</b>	203. 2	177. 02	203. 2
<b>Mean</b>	176. 221	167. 054	176. 201
<b>Std Dev</b>	9. 22544	14. 0951	9. 23729

### 3.4. Retrieving Multiple Imputed Data Sets

The IMPUTE module outputs a single data set, the one specified on the DATAOUT statement of your setup file. If you have requested more than one imputation with the keyword MULTIPLE and have included the keyword ALL in the DATAOUT statement the imputations are concatenated in the output file. The imputations can be distinguished by the system variable `_MULT_`.

If you request more than one imputation with the keyword MULTIPLE and have not included the keyword ALL in DATAOUT statement only the first imputation will be included in the output file. The additional imputations can be retrieved by submitting the `%PUTDATA` macro statement in the SAS program editor:

For example,

Setup IMPSETUP (see 3.2) requested 2 imputations but does not include ALL in the DATAOUT statement. To retrieve the second imputed data set you would submit the following commands in the SAS program editor:

```
LIBNAME MYOUT 'C:\MYOUTDIR';  
%PUTDATA(NAME=MYSETUP, DIR=C:\MYOUTDIR, MULT=2, DATAOUT=MYOUT. IMPDATA2);  
RUN;
```

In this example,

The name **MYOUT** is assigned to the directory C:\MYOUTDIR with the SAS Libname statement.

The `%PUTDATA` macro retrieves additional imputed data sets. The keywords within the parentheses contain information about the imputed data set to be retrieved.

**Name**= references the setup file used in the initial imputation (mysetup).

**Dir**= specifies the directory where the setup is located (c:\myoutdir).

**Mult**= indicates which multiple imputation to retrieve (2).

**Dataout**= assigns a storage location (MYOUT) and a file name (IMPDATA2) to the retrieved imputed data set.

## 4. DESCRIBE

The DESCRIBE module estimates population means, proportions, subgroup differences, contrasts and linear combinations of means and proportions. A Taylor Series approach is used to obtain variance estimates under complex sample designs. A multiple imputation analysis can be performed using the DESCRIBE module.

### 4.1. DESCRIBE Statements

#### Required or Standard Statements

**DATAIN libname.filename;**

This keyword identifies the location and name of the SAS data set to be analyzed.

For example,

**DATAIN Mylib.Mydata;**

indicates that the SAS data file Mydata is located in the library Mylib. Mylib is the name assigned to a directory with the SAS Libname statement. (See section 2 for examples of SAS Libname statements or consult the SAS user manual for a more extensive discussion of Libname.)

To perform multiple imputation analysis, more than one SAS data file can follow the DATAIN keyword in the DESCRIBE module. When multiple data sets are specified each is analyzed separately and the inferences--estimates and variances--are combined (Rubin 1987b).

For example, imputation setup MYSETUP (see 3.2) requested two imputations of the SAS file MYDATA. The resulting imputed data sets IMPDATA and IMPDATA2 (see 3.4) can be listed on the DATAIN statement as follows:

**DATAIN Mylib.Impdata Mylib.Impdata2;**

**RUN;**

This should be the last statement in the setup file.

#### Optional Statements

**STRATUM variable name;**

**variable name** is the name of the stratum variable. No missing values are allowed for the stratum variable.

**CLUSTER variable name;**

**variable name** is the Primary Sampling Unit (PSU) or Sampling Error Computing Unit (SECU) variable. No missing values are allowed for the cluster variable.

**WEIGHT variable name;**

**variable name** is the sampling weight variable. Sampling weights are usually the product of selection, nonresponse adjustment and poststratification weights. No missing values are allowed for the weight variable.

**MODEL method;**

MODEL indicates the variance estimation method to be used. **Mult** (Default) is useful when there are multiple PSUs within a stratum, **Pair** employs the paired selection method, and **Diff** employs the successive differences method. You can specify different methods for each stratum.

For example,

**MODEL Pair(15, 16, 17) Diff(20, 21, 27);**

will use paired differences for strata 15, 16, 17, the successive differences for strata 20, 21,27, and **Mult** for the rest.

**TABLE variable list;**

This command will produce the weighted proportions and their standard errors for all levels of a variable(s).

**TABLE Race;**

Crosstabulations may be indicated with an asterisk, for example:

**TABLE Race\*Gender;**

**MEAN variable list;**

Means, standard errors, and design effects are calculated for the list of variables following the MEAN keyword.

For example,

**MEAN BMI Age;**

will compute the means of BMI and Age.



On the other hand,

**MEAN Var1-Var20;**

will compute the means of all the variables between the "locations" of Var1 and Var20 in the SAS data set.

**BY list;**

The BY keyword is used in conjunction with the TABLE or MEAN keyword. The analyses will be performed for each level of the variable(s) specified in the BY statement.

For instance,

**TABLE Race;**

**BY Gender;**

will produce the weighted proportion of each Race category for each of the two levels of Gender.

If variable Agecat is age in 3 categories then

**TABLE Race;**

**BY Gender Agecat;**

will produce weighted proportion of each Race category for each of the six combinations of Gender and Agecat.

**CONTRAST specifications;**

CONTRAST is used in conjunction with the MEAN keyword to compare or estimate linear combinations of cell means or proportions.

For example,

**MEAN Income;**

**CONTRAST Race;**

will produce all the pairwise comparisons of mean Income defined by Race. If Race has three categories then three pairwise comparisons will be produced.

**MEAN Income;**

**CONTRAST Race\*Gender;**

will produce comparisons of Income means for all combinations of Race and Gender.

**MEAN Income;**  
**CONTRAST Race (. 33 . 33 . 33);**

will produce the average across the three categories of Race. (If Race has more than three levels then the above statement will produce an error message).

You can also specify complicated statements such as

**MEAN Income;**  
**CONTRAST Race(- 1 0 1)\*Gender(- 1 1);**

This can be useful in testing the significance of some preplanned contrasts in an ANOVA setting.

**NOBS number;**

NOBS indicates the number of observations to be used in the analysis. By default all observations in the data set will be used. You might use NOBS to subset a large data set while testing your setup file.

**MDATA instruction;**

The keyword MDATA followed by an instruction (STOP/IMPUTE/SKIP) indicates how missing data should be treated by the DESCRIBE module. If MDATA is not included in your setup, cases with missing data will be excluded from your analysis. This is equivalent to using the SKIP instruction.

**MDATA STOP;**

will cause the DESCRIBE module to stop if missing data are encountered in any analysis variables.

**MDATA IMPUTE;**

will impute missing data when used in conjunction with IMPUTE keywords. (See section 4.4 for more on combining IMPUTE and DESCRIBE functions).

**PRINT instruction;**

Indicates the printout desired. The options are STANDARD (default) and DETAILS.

When a DESCRIBE procedure includes the IMPUTE missing-data option (see MDATA above) the DETAILS keyword instructs *IVEware* to print the number and distribution of observed values, imputed values, and combined observed and imputed values for each variable.

When a DESCRIBE procedure includes multiple imputations the DETAILS keyword instructs *IVEware* to print estimates and statistics for each imputed data set as well as combined estimates and statistics across the imputed data sets.

The STANDARD DESCRIBE printout does not include imputation results.

**TITLE text \n text;**

Indicates the title(s) to be printed at the top of each page of the printout. A \n indicates that the text that follows should be printed on the next line.

For example,

**TITLE This is the title on the first line \n This is the title on the second line;**

**4.2. DESCRIBE Setup File**

The following is an example of a DESCRIBE setup file.<sup>2</sup>

```
LIBNAME MYLIB 'C:\MYINDIR';

%DESCRIBE(NAME=DESCEG, DIR=C:\MYOUTDIR, SETUP=NEW);
  DATAIN      MYLIB.MYDATA;
  STRATUM      STRATUM;
  CLUSTER      PSU;
  WEIGHT       FNLWGT2;
  MODEL        MULT;
  MEAN         AGE84 HEIGHT WEIGHT;
  CONTRAST     SEX;
RUN;
```

**4.3. DESCRIBE List File**

The above setup file produces the following in the SAS output window which is also stored in a file called **DESCEG.LST**.

IVEware Setup Checker, Wed Sep 19 13:17:40 2001

1

Setup listing:

```
DATAIN      MYLIB.MYDATA;
STRATUM      STRATUM;
CLUSTER      PSU;
WEIGHT       FNLWGT2;
MODEL        MULT;
MEAN         AGE84 HEIGHT WEIGHT;
CONTRAST     SEX;
RUN;
```

IVEware Design-Based Descriptive Statistics Procedure, Wed Sep 19 13:17:41 2001

1

```
Stratum variable  STRATUM
Secu variable     PSU
Weight variable   FNLWGT2
```

<sup>2</sup> This setup is included in the example files available at [www.isr.umich.edu/src/smp/ive/](http://www.isr.umich.edu/src/smp/ive/). See DESCRIBE.SAS.

Analysis description:

7 Variables  
108 Strata  
216 Secus

Strata Model  
108 Multiple PSU  
0 Paired Selection  
0 Successive Differences

5151 Cases Read

IVEware Design-Based Descriptive Statistics Procedure, Wed Sep 19 13:17:41 2001 2

Problem 1

Degrees of freedom  
108

Factor Covariance of denominator  
SEX 0.02850  
1

Factor	Number of Cases	Sum of Weights	Weighted Mean	Standard Error
AGE84	1856	6710914	76.345178	0.097017591

Lower Bound	Upper Bound	T Test	Prob >  T
76.152872	76.537484	786.92099	0.00000

Unweighted Mean	Bias	Design Effect
77.62069	1.67072	0.63188

Factor	Covariance of denominator			
SEX	0.02815			
2				
Mean	Number of	Sum of	Weighted	Standard
AGE84	Cases	Weights	Mean	Error
	3295	10623950	77.203558	0.11008757
	Lower	Upper	T Test	Prob >  T
	Bound	Bound		
	76.985345	77.421771	701.29227	0.00000
	Unweighted	Bias	Design	
	Mean		Effect	
	78.537481	1.72780	1.21811	

Contrast  
SEX  
1 versus  
2

Mean	Number of	Sum of	Weighted	Standard
AGE84	Cases	Weights	Mean	Error
	5151	17334864	-0.85838011	0.14180786
	Lower	Upper	T Test	Prob >  T
	Bound	Bound		
	-1.1394685	-0.57729177	-6.05312	0.00000
	Unweighted	Bias	Design	
	Mean		Effect	
	-0.91679138	6.80482	0.80939	

**Problem 2**

Degrees of freedom

108

Factor Covariance of denominator

SEX 0.02877

1

Mean	Number of	Sum of	Weighted	Standard
HEIGHT	Cases	Weights	Mean	Error
	1844	6679293	68.585631	0.068650078

Lower	Upper	T Test	Prob >  T
Bound	Bound		
68.449555	68.721708	999.06123	0.00000

Unweighted	Bi as	Desi gn
Mean		Effect
68.460954	-0.18178	0.99954

Factor Covariance of denominator

SEX 0.02855

2

Mean	Number of	Sum of	Weighted	Standard
HEIGHT	Cases	Weights	Mean	Error
	3257	10506114	63.11827	0.052626322

Lower	Upper	T Test	Prob >  T
Bound	Bound		
63.013955	63.222585	1199.36694	0.00000

Unweighted	Bi as	Desi gn
Mean		Effect

	63. 07338	- 0. 07112	1. 19678	
Contrast				
SEX				
1 versus				
2				
Mean	Number of	Sum of	Weighted	Standard
HEIGHT	Cases	Weights	Mean	Error
	5101	17185407	5. 4673611	0. 089163514
	Lower	Upper	T Test	Prob >  T
	Bound	Bound		
	5. 2906232	5. 644099	61. 31837	0. 00000
	Unweighted	Bi as	Design	
	Mean		Effect	
	5. 387574	- 1. 45933	1. 13102	

IVEware Design-Based Descriptive Statistics Procedure, Wed Sep 19 13:17:41 2001 4

**Problem 3**

Degrees of freedom  
108

Factor Covariance of denominator  
SEX 0. 02864  
1

Mean	Number of	Sum of	Weighted	Standard
WEIGHT	Cases	Weights	Mean	Error
	1842	6667794	166. 0287	0. 59423655
	Lower	Upper	T Test	Prob >  T
	Bound	Bound		
	164. 85081	167. 20658	279. 39832	0. 00000



	Unwei ghted Mean	Bi as	Desi gn Effect	
	164. 47448	- 0. 93611	0. 91722	
Factor	Covari ance of denomi nator			
SEX	0. 02846			
2				
Mean WEI GHT	Number of Cases	Sum of Wei ghts	Wei ghted Mean	Standard Error
	3244	10467704	138. 90897	0. 49371546
	Lower Bound	Upper Bound	T Test	Prob >  T
	137. 93034	139. 8876	281. 35431	0. 00000
	Unwei ghted Mean	Bi as	Desi gn Effect	
	137. 70962	- 0. 86341	1. 04083	
Contrast				
SEX	1 versus			
2				
Mean WEI GHT	Number of Cases	Sum of Wei ghts	Wei ghted Mean	Standard Error
	5086	17135498	27. 119725	0. 73043267
	Lower Bound	Upper Bound	T Test	Prob >  T
	25. 671878	28. 567572	37. 12830	0. 00000
	Unwei ghted Mean	Bi as	Desi gn Effect	
	26. 764867	- 1. 30849	0. 86167	

#### 4.4. IMPUTE/DESCRIBE Combined

The DESCRIBE module permits multiple imputation of missing data prior to the execution of a DESCRIBE analysis. To perform a combined DESCRIBE/IMPUTE procedure your setup file must include the keyword –instruction statement MDATA IMPUTE. A combined DESCRIBE/IMPUTE setup file can include any or all of the IMPUTE keywords defined in section 3.

The following is an example of a combined DESCRIBE/IMPUTE setup file.<sup>3</sup>

```
LIBNAME MYLIB 'C:\MYINDIR';

%DESCRIBE (SETUP=NEW, NAME=DISETUP, DIR=C:\MYOUTDIR);
  DATA IN MYLIB.MYDATA;
  MDATA IMPUTE;

  *Impute keywords;
  DEFAULT TRANSFER;
  CONTINUOUS AGE84 HEIGHT WEIGHT;
  CATEGORICAL SEX;
  ITERATIONS 10;
  MULTIPLES 5;
  SEED 100;

  *Describe keywords;
  STRATUM STRATUM;
  CLUSTER PSU;
  WEIGHT FNLWGT2;
  MODEL MULT;
  MEAN AGE84 HEIGHT WEIGHT;
  CONTRAST SEX;
RUN;
```

The above setup file multiply imputes the missing values for the 4 variables (AGE84, HEIGHT WEIGHT, SEX) before performing the DESCRIBE analysis.

The multiple imputation analysis involves repeating the same analysis on each imputed data set and then combining the point estimates and the variances using the formula given in Rubin (1987b) and Li, Raghunathan and Rubin (1991). Suppose that  $M$  is the number of imputations (in our setup file  $M=5$ ) and  $e_l$  are the estimates from the imputed data set  $l=1,2,\dots,M$ . Let  $v_l$  be the corresponding variances (square of the standard errors) of the estimates. The multiply imputed estimate is

---

<sup>3</sup> This setup is included in the example files available at [www.isr.umich.edu/src/smp/ive/](http://www.isr.umich.edu/src/smp/ive/). See IMP\_DESC.SAS.

$$e_{MI} = \frac{1}{M} \sum_{l=1}^M e_l$$

and the multiply imputed variance estimate is

$$v_{MI} = \bar{v}_M + \frac{M+1}{M} B_M$$

where

$$\bar{v}_M = \frac{1}{M} \sum_{l=1}^M v_l$$

and

$$B_M = \frac{1}{M-1} \sum_{l=1}^M (e_l - e_{MI})^2$$

The confidence intervals are constructed using a  $t$  reference distribution with degrees of freedom,

$$n = (M-1)(1 + 1/\bar{r}_M)^2$$

where

$$\bar{r}_M = \frac{M+1}{M} \frac{B_M}{\bar{v}_M}$$

#### 4.4.1. Better Analysis with Incomplete Data

Most theoretical and empirical investigations suggest that the creation of imputed values should be conditional on as many observed variables as possible. The DESCRIBE setup file given in the previous section, however, only employed a narrow set of variables in the imputation procedure, the four variables of analytical interests--AGE84, HEIGHT, WEIGHT, and SEX. A preferable approach is to impute the missing values of the four variables by using other variables in the data set as auxiliary variables before invoking the DESCRIBE macro. The following setup file performs imputations, extracts multiply imputed data sets and then analyze the imputed data sets.

```
LIBNAME MYLIB 'C:\MYINDIR';
LIBNAME MYOUT 'C:\MYOUTDIR';

/* First multiply impute all the missing values in the input data set */
%IMPUTE(NAME=MYSETUP, DIR=C:\MYOUTDIR, SETUP=NEW);
  DATAIN      MYLIB.MYDATA;
  DATAOUT     MYOUT.IMPUTED1;
  MULTIPLES    5;
  Other IMPUTE keywords
RUN;

/* Extract the 5 imputed data sets */
%PUTDATA(NAME=MYSETUP, DIR=C:\MYOUTDIR, MULT=2, DATAOUT=MYOUT.IMPUTED2); RUN;
%PUTDATA(NAME=MYSETUP, DIR=C:\MYOUTDIR, MULT=3, DATAOUT=MYOUT.IMPUTED3); RUN;
%PUTDATA(NAME=MYSETUP, DIR=C:\MYOUTDIR, MULT=4, DATAOUT=MYOUT.IMPUTED4); RUN;
%PUTDATA(NAME=MYSETUP, DIR=C:\MYOUTDIR, MULT=5, DATAOUT=MYOUT.IMPUTED5); RUN;

/* Analyze the multiply imputed data sets */
%DESCRIBE (SETUP=NEW, NAME=DISETUP, DIR=C:\MYOUTDIR);
  DATAIN      MYOUT.IMPUTED1
               MYOUT.IMPUTED2
               MYOUT.IMPUTED3
               MYOUT.IMPUTED4
               MYOUT.IMPUTED5;
  STRATUM     STRATUM;
  CLUSTER     PSU;
  WEIGHT      FNLWGT2;
  MODEL       MULT;
  MEAN        AGE84 HEIGHT WEIGHT;
  CONTRAST    SEX;
RUN;
```

## 5. REGRESS

The REGRESS module fits linear, logistic, Poisson, polytomous and proportional hazards regression models for data resulting from complex sample designs. The Jackknife Repeated Replication technique is used to estimate variances (Kish and Frankel 1974).

### 5.1. REGRESS Statements

#### Required or Standard Statements

##### **DATAIN libname.filename;**

This keyword identifies the location and name of the SAS data set to be analyzed.

For example,

##### **DATAIN Mylib.Mydata;**

indicates that the SAS data file Mydata is located in the library Mylib. Mylib is the name assigned to a directory with the SAS Libname statement. (See section 2 for examples of SAS Libname statements or consult the SAS user manual for a more extensive discussion of Libname.)

More than one SAS data file can follow the DATAIN keyword in the DESCRIBE module. The use of more than one data file is restricted to the analysis of a multiply imputed data file. When multiple data sets are specified each is analyzed separately and the inferences--estimates and variances--are combined (Rubin 1987b).

For example, imputation setup MYSETUP (see 3.2) requested two imputations of the SAS file MYDATA. The resulting imputed data sets IMPDATA and IMPDATA2 (see 3.4) can be listed on the DATAIN statement as follows:

##### **DATAIN Mylib.Impdata Mylib.Impdata2;**

##### **DEPENDENT variable name;**

This specifies the name of the dependent variable in the regression model. Dependent variables are assumed to be continuous unless the CATEGORICAL keyword is included in the setup file (see below).

##### **PREDICTOR variable list;**

This specifies the right hand side of the regression model. Predictor variables are assumed to be continuous unless they are defined as CATEGORICAL (see below). Interaction terms can be specified by using the "\*" notation.

For example,

**PREDICTOR** *Income* *Age* *Income\*Age*;

**LINK model;**

LINK defines the type of regression model to be fit. Specify **Linear** for fitting a multiple linear regression model, **Logistic** for fitting a logistic (binary) or generalized logistic (polytomous) regression model, **Log** for fitting a Poisson regression model for a count variable, **Tobit** for fitting a tobit model or **Phreg** for fitting Proportional Hazards model (Cox model).

**RUN;**

This should be the last statement in the setup file.

**Optional Statements**

**STRATUM variable name;**

**variable name** is the name of the stratum variable. No missing values are allowed for the stratum variable.

**CLUSTER variable name;**

**variable name** is the Primary Sampling Unit (PSU) or Sampling Error Computing Unit (SECU) variable. No missing values are allowed for the cluster variable.

**WEIGHT variable name;**

**variable name** is the survey weight variable. Survey weights are usually the product of selection, nonresponse adjustment and poststratification weights. No missing values are allowed for the weight variable.

**NOTES:**

1. If the STRATUM, CLUSTER and WEIGHT variable are not specified then a simple random sample analysis will be performed.
2. If a design based analysis involves only a WEIGHT variable and no STRATUM or CLUSTER variable then a pseudo stratification variable and a pseudo cluster variable should be used. When using pseudo variables, all observations in the data set should have the same value for the pseudo STRATUM variable (e.g., 1), while each observation should have a unique value on the pseudo CLUSTER variable (e.g., observation ID number or SAS system variable `_N_`). The pseudo variables should be created in a SAS data step prior to performing the analysis. See the Appendix for an example data step creating pseudo stratification and pseudo cluster variables. The inclusion of pseudo variables will increase the time REGRESS needs for analysis.

**CENSOR variable name (number);**

**variable name** is a censoring variable, and **number** is the code indicating censoring. If the number is omitted then by default 1 will be considered as the code indicating censoring. The Censor statement is required if the **LINK** is specified as **Phreg**.

For example,

```
LINK           Phreg;
DEPENDENT     Survival time;
CENSOR        Died (0);
```

In this example, respondent survival time is predicted censoring on whether or not the respondent died.

**CATEGORICAL variable list;**

declares that the listed variables are to be treated as categorical. If a variable with  $k$  categories is listed on the CATEGORICAL and PREDICTOR statement then  $k-1$  predictors (dummies) will be included in the regression model. The category with the highest code value will be the reference category.

**OFFSETS count variable(offset variable);**

This statement is used to specify an offsets variable when fitting a Poisson regression model.

For example,

```
OFFSETS  Injuries(Years);
```

will fit a model predicting the number for injuries occurring per year.

**ID variable name;**

Specifies the variable to be used as the unique subject identifier. This allows for linking the PREDOUT file (see below) created by the REGRESS module to other files.

**NOINTER;**

This keyword will fit regression models without the intercept term.

**ESTIMATES label: specification;**

This is useful for estimating values of the dependent variable for a specific set of covariates or testing hypotheses involving the estimated regression coefficients.

For example, suppose that the following regression model is fit:

$$Y = b_0 + b_1x_1 + b_2x_2 + b_3x_3$$

If we are interested in predicting  $Y$  when  $x_1 = 1$ ,  $x_2 = 2$  and  $x_3 = 0$  then we can obtain the predicted value and the 95% confidence interval by using the following statement:

```
ESTIMATES   Mylabel : Intercept (1) X1(1) X2(2);
```

Several estimates can be requested by separating them with “/” .

```
ESTIMATES   Mylabel 1: Intercept (1) X1(1) X2(2) /  
            Mylabel 2: Intercept (1) X1(1) X3(1) /  
            Mylabel 3: Intercept (1) X2(1) X3(1) ;
```

**BY variable list;**

The regression analysis will be performed for each level of the variable(s) specified in the BY statement.

For instance,

```
BY Gender;
```

will produce regressions for each of the two levels of Gender.

If the variable Agecat is age in 3 categories then

```
BY Gender Agecat;
```

will produce regressions for each of the six combinations of Gender and Agecat.

**NOBS number;**

NOBS indicates the number of observations to be used in the analysis. By default all observations in the data set will be used. You might use NOBS to subset a large data set while testing your setup file.

**MDATA instruction;**

The keyword MDATA followed by an instruction (STOP/IMPUTE/SKIP) indicates how missing data should be treated by the REGRESS module. If MDATA is not included in your setup cases with missing data for model variables will be excluded from the regression. This is equivalent to using the SKIP instruction.



**MDATA Stop;**

will cause the REGRESS module to stop if missing data are encountered for variables in the regression model.

**MDATA Impute;**

will impute missing data when used in conjunction with IMPUTE keywords. See section 5.4 for more on combining IMPUTE and REGRESS functions.

**PLOT libname.filename;**

This keyword creates a series of diagnostic plots including residual, leverage, influence and normal probability plots. The plots will be stored in the file name specified after the PLOT keyword.

**MAXLOGI number;**

Specifies the maximum number of iterative algorithms to be performed in a logistic or multilogit regression model. The default is 50. This is useful if the Newton-Raphson algorithm used in producing maximum likelihood estimates does not converge after 50 iterations. This applies to the convergence criterion for the logistic, polytomous and Poisson regression models. You can check whether you have a non-convergence problem by inspecting the log file (e.g.,mysetup.log).

**MINCODI decimal;**

Specifies the minimum proportional change in any regression coefficient to continue the logistic regression iteration process. This applies to the convergence criterion for the logistic, polytomous and Poisson regression models.

**PRINT instruction;**

Indicates the printout desired. The options are STANDARD (default) and DETAILS.

When a REGRESS procedure includes the IMPUTE missing-data option (see MDATA above) the DETAILS keyword instructs *IVEware* to print the number and distribution of observed values, imputed values, and combined observed and imputed values for each variable.

When a REGRESS procedure includes multiple imputations the DETAILS keyword instructs *IVEware* to print the estimates and statistics for each multiple as well as for the combined multiples.

The STANDARD REGRESS printout does not include imputation results.

**TITLE text \n text;**

Indicates the title(s) to be printed at the top of each page of the printout. A \n indicates that the text that follows should be printed on the next line.

For example,

**TITLE This is the title on the first line \n This is the title on the second line;**

**OUTPUT FILES**

REGRESS can output SAS data sets for diagnostic purposes and for further analysis.

**PREDOUT libname.filename;**

outputs a file containing predicted values, residuals and leverage information.

For example,

**PREDOUT Mylib.Predval ;**

creates an output file called **Predval** containing the predicted values, their standard errors and 95% confidence intervals. If an ID statement is included in the setup an ID variable is also included in the data set.

**ESTOUT libname.filename;**

Outputs a file containing estimates and their variances-covariances. (ESTOUT produces a SAS estimate type data set. See the SAS user manual for more details).

For example,

**ESTOUT Mylib.Est;**

creates an output file called **Est** containing the estimates and variances-covariances.

**REPOUT libname.filename;**

Outputs a file containing estimates for each replicate. Estimated regression coefficients are provided for each combination of STRATUM, CLUSTER and BY variable.

**5.2. REGRESS Setup File**

The following is an example of a REGRESS setup file.<sup>4</sup>

```
LIBNAME MYLIB 'C:\MYINDIR';

%REGRESS (SETUP=NEW, NAME=REGSETUP, DIR=C:\MYINDIR);
  DATAIN      MYLIB.MYDATA;
  ESTOUT       MYLIB.EST;
  REPOUT       MYLIB.REP;
  PLOTS        MYLIB.MYPLOTS;
  WEIGHT       FNLWGT2;
  STRATUM      STRATUM;
  CLUSTER      PPSU;
  MDATA        SKIP;
  LINK         LINEAR;
  CATEGORICAL HEALTH;
  DEPENDENT    DV12;
  PREDICTOR    AGE84 EDUC SEX HEALTH;
  ESTIMATES
    AGEX:      INTERCPT (1) AGE84 (1)      /
    EDUC:      INTERCPT (1) EDUC (1)       /
    HEALTH1:   INTERCPT (1) HEALTH (1)     /
    HEALTH2:   INTERCPT (1) HEALTH (0 1)   /
    HEALTH3:   INTERCPT (1) HEALTH (0 0 1) /
    HEALTH4:   INTERCPT (1) HEALTH (0 0 0 1);
RUN;
```

---

<sup>4</sup> This setup is included in the example files available at [www.isr.umich.edu/src/smp/ive/](http://www.isr.umich.edu/src/smp/ive/). See REGRESS.SAS.

**5.3. REGRESS List File**

The above setup file produces the following in the SAS output window.

IVEware Setup Checker, Wed Sep 19 13: 30: 15 2001 1

Setup listing:

```

DATAIN      MYLIB. MYDATA;
ESTOUT      MYLIB. EST;
REPOUT     MYLIB. REP;
PLOT       MYLIB. MYPLOTS;
WEIGHT     FNLWGT2;
STRATUM    STRATUM;
CLUSTER    PPSU;
MATA      SKIP;
LINK      LINEAR;
CATEGORI CAL HEALTH;
DEPENDENT DV12;
PREDICTOR  AGE84 EDUC SEX HEALTH;
ESTIMATES
  AGEX:    INTERCPT (1) AGE84 (1) /
  EDUC:    INTERCPT (1) EDUC (1) /
  HEALTH1: INTERCPT (1) HEALTH (1) /
  HEALTH2: INTERCPT (1) HEALTH (0 1) /
  HEALTH3: INTERCPT (1) HEALTH (0 0 1) /
  HEALTH4: INTERCPT (1) HEALTH (0 0 0 1);
RUN;

```

IVEware Jackknife Regression Procedure, Wed Sep 19 13: 30: 17 2001 1

```

Regression type:      Linear
Dependent variable:  DV12      Doctor visits in past 12 months
Predictors:          AGE84      Age during 1984
                   EDUC        Education of individual - completed years
                   SEX
                   HEALTH      Health status
Cat. var. ref. codes: HEALTH    5
Stratum variable:    STRATUM
Cluster variable:    PPSU        Pseudo PSU codes
Weight variable:     FNLWGT2     Final 1986 LSOA weight

```

Number of valid cases: 5001  
 Sum of weights: 16887191  
 Number of replicates: 108

Degrees of freedom: 108

Sum of squares:

Model: 133654133.22  
 Error: 3552783493.5  
 Total: 3686437626.7  
 R-square: 0.03626  
 F-value: 0.50786  
 P-value: 0.84818

Variable	Estimate	Jackknife Std Error	T Test	Prob >  T
INTERCPT	9.6785585	2.6780762	3.61400	0.00046
AGE84	0.0242635	0.0251731	0.96387	0.33727
EDUC	0.1172407	0.0782709	1.49788	0.13708
SEX	0.1653307	0.5274063	0.31348	0.75452
HEALTH.1	-9.3422188	1.1561745	-8.08028	0.00000
HEALTH.2	-9.1956046	0.9648200	-9.53090	0.00000
HEALTH.3	-7.7243830	1.1512202	-6.70974	0.00000
HEALTH.4	-5.4633127	0.9477212	-5.76468	0.00000

Variable	Estimate	95% Confidence Interval	
		Lower	Upper
INTERCPT	9.6785585	4.3701365	14.9869806
AGE84	0.0242635	-0.0256340	0.0741609
EDUC	0.1172407	-0.0379061	0.2723876
SEX	0.1653307	-0.8800821	1.2107435
HEALTH.1	-9.3422188	-11.6339617	-7.0504759
HEALTH.2	-9.1956046	-11.1080489	-7.2831603
HEALTH.3	-7.7243830	-10.0063057	-5.4424603
HEALTH.4	-5.4633127	-7.3418643	-3.5847611

Variable	Design Effect	SRS Estimate	% Diff SRS v Est
INTERCPT	0.86585	9.9872262	3.18919
AGE84	0.53647	0.0093774	-61.35179

EDUC	1. 94152	0. 1289522	9. 98921
SEX	1. 54705	0. 4928169	198. 07952

IVEware Jackknife Regression Procedure, Wed Sep 19 13: 30: 17 2001 2

Variable	Design Effect	SRS Estimate	% Diff SRS v Est
HEALTH. 1	2. 12508	-9. 1466473	-2. 09342
HEALTH. 2	1. 69092	-8. 9212334	-2. 98372
HEALTH. 3	2. 76780	-7. 6200584	-1. 35059
HEALTH. 4	1. 70318	-5. 0605819	-7. 37155

IVEware Jackknife Regression Procedure, Wed Sep 19 13:30:17 2001

3

Covariance of Estimates

	INTERCPT	AGE84	EDUC	SEX	HEALTH. 1	HEALTH. 2
INTERCPT	7.1720919709	-0.054874219636	-0.11671756869	-0.6108644999	-0.90663923841	-0.65196961278
AGE84	-0.054874219636	0.000633682999	0.00052840826056	0.00033915018911	0.0013455680422	-0.0010872478134
EDUC	-0.11671756869	0.00052840826056	0.0061263368185	0.014597546655	-0.00082235432866	-0.014289667761
SEX	-0.6108644999	0.00033915018911	0.014597546655	0.27815738064	-0.04883860057	0.0031860847961
HEALTH. 1	-0.90663923841	0.0013455680422	-0.00082235432866	-0.04883860057	1.3367394257	0.8782781356
HEALTH. 2	-0.65196961278	-0.0010872478134	-0.014289667761	0.0031860847961	0.8782781356	0.93087758378
HEALTH. 3	-0.14583706877	-0.0015140019441	-0.034258007379	-0.20983014239	1.003215425	0.9558855169
HEALTH. 4	-1.1410723126	0.0033713479862	0.0068841548171	0.027309952996	0.79209352269	0.74359764199

	HEALTH. 3	HEALTH. 4
INTERCPT	-0.14583706877	-1.1410723126
AGE84	-0.0015140019441	0.0033713479862
EDUC	-0.034258007379	0.0068841548171
SEX	-0.20983014239	0.027309952996
HEALTH. 1	1.003215425	0.79209352269
HEALTH. 2	0.9558855169	0.74359764199
HEALTH. 3	1.3253080099	0.81879302712
HEALTH. 4	0.81879302712	0.89817552299

**5.3.1. Estimates Statement Output**

The ESTIMATES statement produces the following results.

IVEware Jackknife Regression Procedure, Mon Dec 17 14:40:28 2001 4

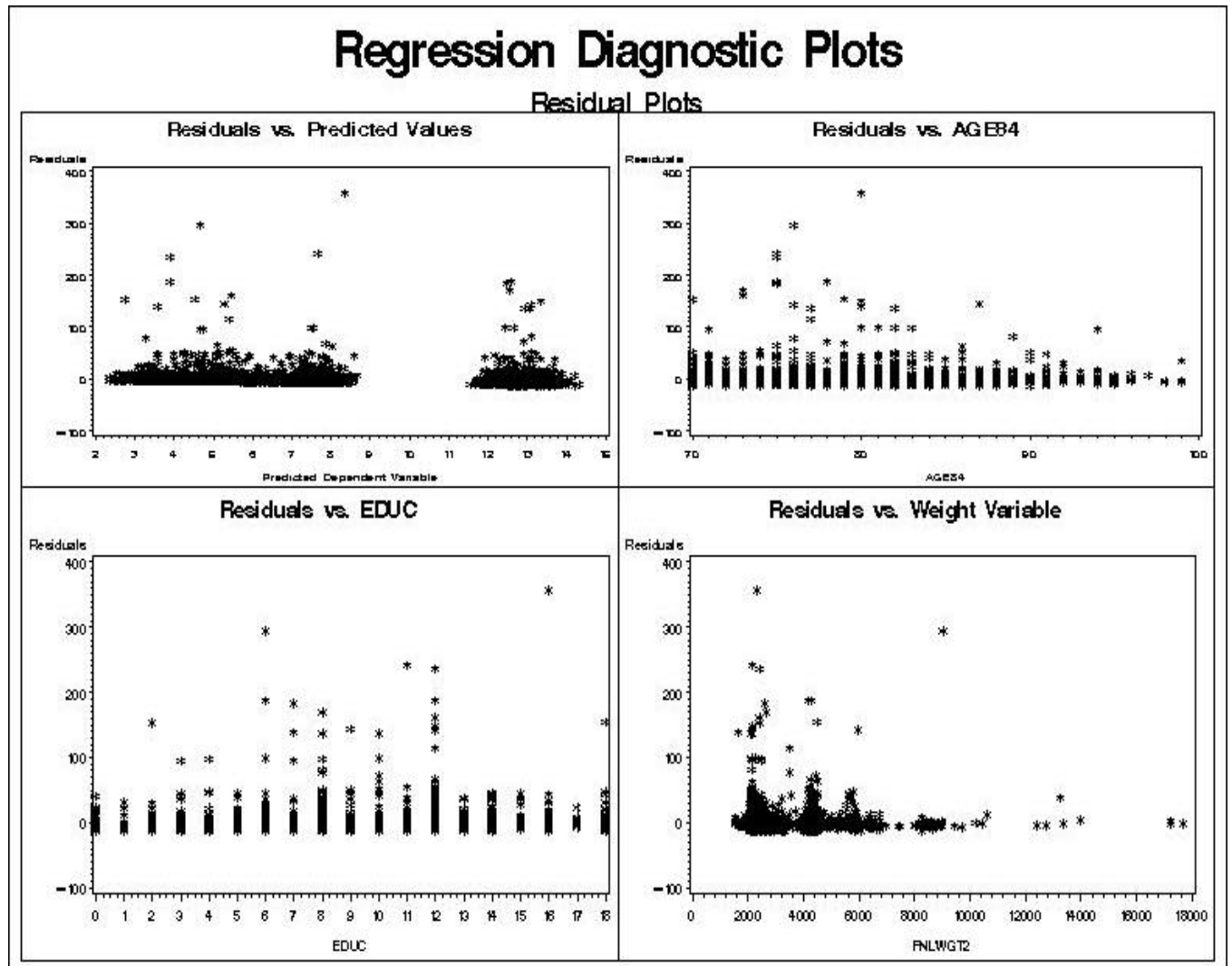
**Estimate calculations**

<b>Name</b>	<b>Estimate</b>	<b>Standard Error</b>	<b>Design Effect</b>
AGEX	9.7028220	2.6576262	0.87213
EDUC	9.7957993	2.6352957	0.84643
HEALTH1	0.3363397	2.5875767	0.79974
HEALTH2	0.4829539	2.6074950	0.81773
HEALTH3	1.9541755	2.8645638	1.01222
HEALTH4	4.2152458	2.4058518	0.70989



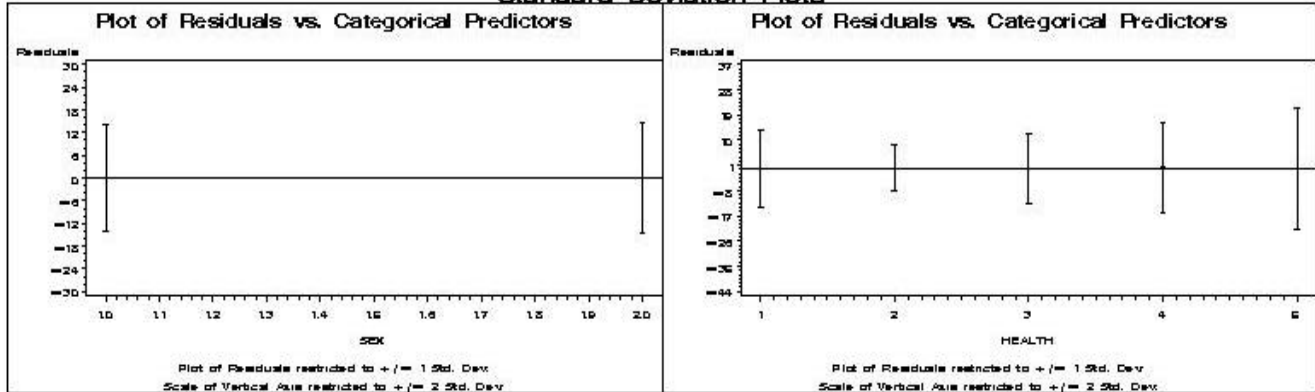
**5.3.2. Plot Statement Output**

The following diagnostic plots are created and stored in MYPLOTS.



# Regression Diagnostic Plots

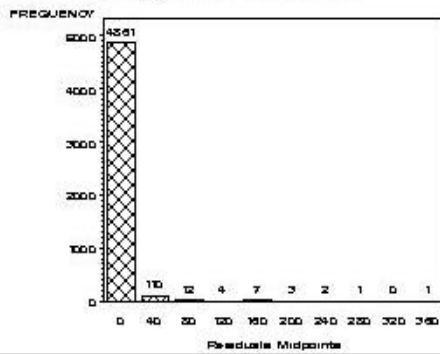
## Standard Deviation Plots



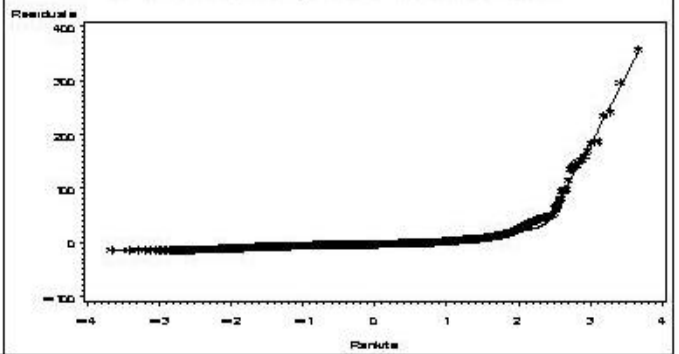
# Regression Diagnostic Plots

## Histograms and Normal Probability Plot

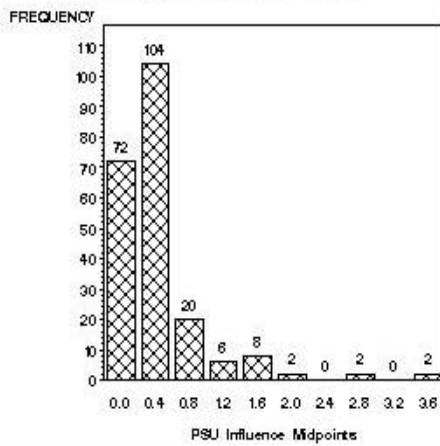
Histogram of Residuals



Normal Probability Plot of the Residuals



Histogram of Influences



**5.4. IMPUTE/REGRESS Combined**

Although we strongly recommend that users impute missing data separately and for the entire data set using the IMPUTE module prior to executing a REGRESS analysis it is possible to use a single setup file to impute missing data and fit a REGRESS regression..

To perform a combined IMPUTE/REGRES procedure your setup file must include the keyword-instruction MDATA Impute. A combined IMPUTE/REGRESS setup file can include any or all IMPUTE keywords described in section 3.1. If MDATA Impute is included in the setup file without any IMPUTE keywords then only the variables in the regression model will be imputed and they will be treated by default as continuous. (Dependent variables in logistic and Poisson regressions, however, will be treated, respectively, as categorical and count variables.)

If you do not include MDATA Impute in your setup file all cases with missing data for model variables will be excluded from the regression. (See section 5.1 for more on the keyword MDATA.)

The following is an example of a combined IMPUTE/REGRESS setup file.

```
LIBNAME MYLIB 'C:\MYINDIR';
%REGRESS (SETUP=NEW, NAME=REGSETUP, DIR=C:\MYDIR);
    DATAIN          MYLIB.MYDATA;
    MDATA            IMPUTE;
    DEFAULT          CONTINUOUS;
    CATEGORICAL      HEALTH SEX POVERTY MARSTAT;
    TRANSFER         STRATUM PPSU FNLWGT2;
    BOUNDS           BDDAY12 (>=0)
                   DV12    (>=0)
                   EDUC    (>=0, <=18)
                   NUMSONS (>=0)
                   NUMDAUGH (>=0);
    ITERATIONS       5;
    MULTIPLES        5;
    SEED             1999;
    ESTOUT           MYLIB.EST;
    WEIGHT           FNLWGT2;
    STRATUM          STRATUM;
    CLUSTER          PPSU;
    LINK             LINEAR;
    DEPENDENT        BDDAY12;
    PREDICTOR        AGE84  EDUC  SEX  POVERTY
                   DV12  HEALTH MARSTAT NUMSONS NUMDAUGH ;
RUN;
```

### 5.4.1. Better Analysis with Incomplete Data

Most theoretical and empirical investigations suggest that the creation of imputed values should be conditional on as many observed variables as possible. The setup file given in the previous section, however, only focused on the narrow set of variables involved in the REGRESS analysis. A preferable approach is to impute the missing values by using other variables in the data set as auxiliary variables before invoking the REGRESS macro.

#### 5.4.1.1 Better Analysis with Incomplete Data--Setup File

The following setup file performs imputations; extracts multiply imputed data sets, and then analyze the imputed data sets.<sup>5</sup>

```
LIBNAME MYLIB 'C:\MYINDIR';
DATA TEMP;
SET MYLIB.MYDATA3;
/*RECODE VARIABLES */
IF OWNBUYR=. THEN OWNBUYR=5;
IF MORTGAGE=. THEN MORTGAGE=4;
IF OWNBUYR=5 THEN AMTOWED=0;
IF OWNBUYR=5 THEN PRESVAL=0;
OWN=0;
IF OWNBUYR IN (1, 2) THEN OWN=1;
IF OWNBUYR=. D THEN OWN=. ;
BLACK=0;
IF RACER=2 THEN BLACK=1;
OTHER=0;
IF RACER=3 THEN OTHER=1;
LOGADL=LOG(NUMADL+1);
DIFFADL=LOG(NUMADL2R+1) - LOGADL;
LOGFLMR=LOG(NUMFLMR+1);
LOGIDL2R=LOG(NUMIDL2R+1);
LOGIADL=LOG(NUMIADL+1);
LOGBD12=LOG(BDDAY12+1);
LOGDV12=LOG(DV12+1);
LOGAMTO=LOG(AMTOWED+1);
LOGPRES=LOG(PRESVAL+1);
RUN;
```

<sup>5</sup> This setup is included in the example files available at [www.isr.umich.edu/src/smp/ive/](http://www.isr.umich.edu/src/smp/ive/). See IMP\_REGRESS.SAS.

```
%IMPUTE(SETUP=NEW, NAME=IMPUTEX, DIR=C: \MYINDIR);
  DATAIN          TEMP;
  DATAOUT         TEMP_1;
  DEFAULT          CATEGORICAL;
  TRANSFER         RESPOND NUMADL NUMADL2R NUMFLMR
                  NUMIADL1 NUMIDL2R BDDAY12 DV12 AMTOWED
                  PRESVAL RACER OWNBUYR;
  CONTINUOUS       DIFFADL LOGADL  EDUC    AGE84
                  LOGFLMR LOGIDL2R LOGIADL LOGBD12 LOGDV12
                  LOGAMFO LOGPRES ;
  RESTRICT         LOGAMFO(MORTGAGE=2),
                  LOGPRES(OWN=1);
  BOUNDS           LOGADL (>=0),
                  LOGFLMR (>=0),
                  LOGIDL2R(>=0),
                  LOGIADL (>=0),
                  LOGBD12 (>=0),
                  LOGDV12 (>=0),
                  LOGPRES (> 0),
                  LOGAMFO (> 0);
  ITERATIONS       5;
  MULTIPLES        5;
  SEED             71901;
RUN;

/*EXTRACT THE REMAINING FOUR MULTIPLY IMPUTED DATA SETS */
%PUTDATA(NAME=IMPUTE2, DIR=C: \MYINDIR, MULT=2, DATAOUT=TEMP_2); RUN;
%PUTDATA(NAME=IMPUTE2, DIR=C: \MYINDIR, MULT=3, DATAOUT=TEMP_3); RUN;
%PUTDATA(NAME=IMPUTE2, DIR=C: \MYINDIR, MULT=4, DATAOUT=TEMP_4); RUN;
%PUTDATA(NAME=IMPUTE2, DIR=C: \MYINDIR, MULT=5, DATAOUT=TEMP_5); RUN;

/*ANALYZE THE IMPUTED DATA SETS */
%REGRESS(NAME=TEMP, DIR=C: \MYINDIR, SETUP=NEW);
  TITLE BETTER MULTIPLE IMPUTATION ANALYSIS;
  DATAIN TEMP_1 TEMP_2 TEMP_3 TEMP_4 TEMP_5;
  DEPENDENT DIFFADL;
  PREDICTOR POVERTY OWN EDUC LOGADL AGE84 SEX BLACK OTHER;
RUN;
```

**5.4.1.2 Better Analysis with Incomplete Data--List File**

The above setup file produces the following in the SAS output window.

**Better Multiple Imputation Analysis**

```

Regression type:      Linear
Dependent variable:  diffadl
Predictors:          POVERTY  NHIS poverty index
                    own
                    EDUC      Education of individual - completed years
                    logadl
                    AGE84     Age during 1984
                    SEX        Sex
                    black
                    other
    
```

IVEware REGRESS Regression Procedure, Tue Jul 17 11: 43: 59 2001 2

**Better Multiple Imputation Analysis**

**All imputations**

```

Number of valid cases: 5151
Sum of weights:        5151

Degrees of freedom     7. 6125987315

Sum of squares:
  Model:               227. 39491938
  Error:               1540. 8913089
  Total:               1768. 2862283
  R-square:            0. 12860
  F-value:             0. 14043
  P-value:             0. 99392
    
```

Variable	Estimate	REGRESS Std Error	T Test	Prob >  T
INTERCPT	-0. 9533687	0. 1491329	-6. 39274	0. 00026
POVERTY	0. 0234619	0. 0260319	0. 90127	0. 39506
own	-0. 0458189	0. 0170959	-2. 68011	0. 02919
EDUC	-0. 0084981	0. 0023877	-3. 55909	0. 00804

logadl	-0.3449836	0.0146736	-23.51046	0.00000
AGE84	0.0179481	0.0016256	11.04100	0.00001
SEX	-0.0118802	0.0183972	-0.64576	0.53741
black	0.0803485	0.0280470	2.86478	0.02211
other	-0.1182457	0.0934084	-1.26590	0.24291

Variable	Estimate	95% Confidence Interval	
		Lower	Upper
INTERCPT	-0.9533687	-1.3003386	-0.6063988
POVERTY	0.0234619	-0.0371035	0.0840274
own	-0.0458189	-0.0855938	-0.0060439
EDUC	-0.0084981	-0.0140533	-0.0029429
logadl	-0.3449836	-0.3791230	-0.3108442
AGE84	0.0179481	0.0141660	0.0217302
SEX	-0.0118802	-0.0546828	0.0309225
black	0.0803485	0.0150949	0.1456021
other	-0.1182457	-0.3355678	0.0990765



## 6. SASMOD

SASMOD macro provides a framework for performing complex design analysis, with or without missing data, for a collection of SAS procedures. Currently the following PROCs are available: CALIS, CATMOD, GENMOD, LIFEREG, MIXED, NLIN, PHREG, and PROBIT. This particular macro executes user-specified SAS PROC commands for every replicate and then combines the results to compute the proper complex design sampling variance estimate. If multiple data sets are specified as inputs, it performs the analysis for each data set and then combines the inferences using the multiple imputation formula given in section 4.4.

The setup is similar to other modules except that a combined IMPUTE and SASMOD has not been implemented. The multiple data sets can be specified in the DATAIN statement. The allowable keywords are DATAIN, BY, STRATUM, CLUSTER, WEIGHT, and TITLE.

### 6.1. SASMOD Statements

#### Required or Standard statements

##### **DATAIN libname.filename;**

This keyword identifies the location and name of the SAS data set to be analyzed.

For example,

##### **DATAIN Mylib.Mydata;**

indicates that the SAS data file **Mydata** is located in the library **Mylib**. Mylib is the name assigned to a directory with the SAS Libname statement. (See section 3 for examples of SAS Libname statements or consult the SAS user manual for a more extensive discussion of Libname.)

**NOTE: To use the SASMOD procedure the SAS data file can only include model variables, “by” variables and design variables (weight, stratum and cluster). No other variables are permitted.**

More than one SAS data file can follow the DATAIN keyword in the REGRESS module. The use of more than one data file is restricted to the analysis of a multiply imputed data file. When multiple data sets are specified each is analyzed separately and the inferences--estimates and variances--are combined (Rubin 1987b).

For example, imputation setup IMPETUP (see 3.2) requested two imputations of the SAS file MYDATA. The resulting imputed data sets IMPDATA and IMPDATA2 (see 3.4) can be listed on the DATAIN statement as follows:

```
DATAIN Mylib.Impdata Mylib.Impdata2;
```

IMPUTE should be run before SASMOD.

**RUN;**

This should be the last statement in the setup file.

### **Optional statements**

**STRATUM variable name;**

**variable name** is the name of the stratum variable. No missing values are allowed for the stratum variable.

**CLUSTER variable name;**

**variable name** is the Primary Sampling Unit (PSU) or Sampling Error Computing Unit (SECU) variable. No missing values are allowed for the cluster variable.

**WEIGHT variable name;**

**variable name** is the survey weight variable. Survey weights are usually the product of selection, nonresponse adjustment and poststratification weights. No missing values are allowed for the weight variable.

### **NOTES:**

1. If the STRATUM, CLUSTER and WEIGHT variable are not specified then a simple random sample analysis will be performed.
2. If a design based analysis involves only a WEIGHT variable and no STRATUM or CLUSTER variable then a pseudo stratification variable and a pseudo cluster variable should be used. When using pseudo variables, all observations in the data set should have the same value for the pseudo STRATUM variable (e.g., 1), while each observation should have a unique value on the pseudo CLUSTER variable (e.g., observation ID number or SAS system variable \_N\_). The pseudo variables should be created in a SAS data step prior to performing the analysis. See the Appendix for an example data step creating pseudo stratification and pseudo cluster variables. The inclusion of pseudo variables will increase the time SASMOD needs for analysis.

**BY variable list;**

The regression analysis will be performed for each level of the variable(s) specified in the BY statement.

For instance,

**BY Gender;**

will produce regressions for each of the two levels of Gender.

If the variable Agecat is age in 3 categories then

**BY Gender Agecat;**

will produce regressions for each of the six combinations of Gender and Agecat.

**TITLE text \n text;**

Indicates the title(s) to be printed at the top of each page of the printout. A \n indicates that the text that follows should be printed on the next line.

For example,

**TITLE This is the title on the first line \n This is the title on the second line;**

**SAS statements**

**PROC procedure name;**

**procedure name** is the name of the SAS procedure. The currently implemented procedures are CALIS, CATMOD, GENMOD, LIFEREG, MIXED, NLIN, PHREG, and PROBIT. This statement must follow the SASMOD statements described above, except for RUN.

**Other SAS statements**

can be used as appropriate for the procedure. However, do not use statements that might lead to more than one model or different models in different replicates or multiples. For example, more than one model statement or specifying a stepwise model is not permitted.

## 6.2. SASMOD Setup File

This example of SASMOD executes a multiple imputation complex design analysis using SAS Proc CATMOD.<sup>6</sup>

```
LIBNAME MYLIB 'C:\MYINDIR' ;

%SASMOD(NAME=SASMOD, DIR=C:\MYINDIR, SETUP=NEW) ;
  TITLE   SASMOD WITH CATMOD;
  DATAIN MYLIB.IMPUTE1
          MYLIB.IMPUTE2
          MYLIB.IMPUTE3
          MYLIB.IMPUTE4
          MYLIB.IMPUTE5;
  CLUSTER PSUVAR;
  STRATUM STRVAR;
  WEIGHT  WTVAR;

/*SAS STATEMENTS BEGIN HERE*/
PROC CATMOD;
  DIRECT   POVERTY FEMALE NONWHITE;
  MODEL    HEALTH=POVERTY FEMALE NONWHITE;
RUN;
```

---

<sup>6</sup> This setup is included in the example files available at [www.isr.umich.edu/src/smp/ive/](http://www.isr.umich.edu/src/smp/ive/). See SASMOD.SAS.

### 6.3. SASMOD List File

The above setup file produces the following in the SAS output window.

IVEware Setup Checker, Wed, Jan 9, 2002, 16:37

Setup listing:

```

TITLE          SASMOD WITH CATMOD;
DATAIN         MYLIB.IMPUTE1
              MYLIB.IMPUTE2
              MYLIB.IMPUTE3
              MYLIB.IMPUTE4
              MYLIB.IMPUTE5;
CLUSTER       PSUVAR;
STRATUM       STRVAR;
WEIGHT        WTVAR ;
PROC CATMOD;
  DIRECT      POVERTY FEMALE NONWHITE;
  MODEL       HEALTH=POVERTY FEMALE NONWHITE;
RUN;
    
```

IVEware Multiple Imputation Regression, Wed Jan 09 17:13:54 2002

1

SASMOD WITH CATMOD

All imputations

```

Valid cases          5151
Sum weights          17334864

Degr freedom         13.0089753

-2 LogLike           53667671.36
    
```

Variable	Estimate	Std Error	Wald test	Prob > Chi
[1]Intercept	1.1446770	0.2119711	29.16168	0.00000
[2]Intercept	1.2262227	0.1685165	52.94854	0.00000
[3]Intercept	1.7738286	0.1752098	102.49585	0.00000
[4]Intercept	0.9634404	0.1870025	26.54330	0.00000
[1]POVERTY	-0.7732373	0.1638440	22.27227	0.00000
[2]POVERTY	-0.6131749	0.1286647	22.71170	0.00000
[3]POVERTY	-0.6427827	0.1319704	23.72332	0.00000
[4]POVERTY	-0.3124112	0.1248928	6.25718	0.01237
[1]FEMALE	0.1987584	0.1185328	2.81173	0.09358
[2]FEMALE	0.2080718	0.1042324	3.98493	0.04591
[3]FEMALE	0.0599650	0.1038824	0.33321	0.56378
[4]FEMALE	0.0508650	0.1136086	0.20045	0.65435
[1]NONWHITE	-0.7957686	0.1723890	21.30859	0.00000
[2]NONWHITE	-0.6441292	0.1789564	12.95542	0.00032
[3]NONWHITE	-0.6544711	0.1441769	20.60581	0.00001

[4]NONWHITE	- 0. 2250937	0. 1318210	2. 91580	0. 08772
<b>Variable</b>	<b>Odds Ratio</b>	<b>95% Confidence Interval</b>		
		<b>Lower</b>	<b>Upper</b>	
[1]Intercept	3. 1414266	1. 9872932	4. 9658306	
[2]Intercept	3. 4083311	2. 3683432	4. 9049988	
[3]Intercept	5. 8933737	4. 0363361	8. 6047971	
[4]Intercept	2. 6206971	1. 7497527	3. 9251567	
[1]POVERTY	0. 4615166	0. 3239468	0. 6575078	
[2]POVERTY	0. 5416285	0. 4101967	0. 7151727	
[3]POVERTY	0. 5258272	0. 3953961	0. 6992841	
[4]POVERTY	0. 7316806	0. 5586643	0. 9582796	
[1]FEMALE	1. 2198872	0. 9443126	1. 5758816	
[2]FEMALE	1. 2313016	0. 9830524	1. 5422408	
[3]FEMALE	1. 0617994	0. 8483657	1. 3289293	
[4]FEMALE	1. 0521808	0. 8232015	1. 3448523	
[1]NONWHITE	0. 4512343	0. 3109367	0. 6548356	
[2]NONWHITE	0. 5251196	0. 3567522	0. 7729472	
[3]NONWHITE	0. 5197169	0. 3806312	0. 7096256	
[4]NONWHITE	0. 7984414	0. 6005823	1. 0614842	

IVEware Multiple Imputation Regression, Wed Jan 09 17: 13: 54 2002

2

SASMOD WITH CATMDD

Covariance of Estimates

	[1]Intercept	[2]Intercept	[3]Intercept	[4]Intercept	[1]POVERTY
[1]Intercept	0. 04493175285	0. 02032375197	0. 02205769121	0. 02503592854	-0. 03040566073
[2]Intercept	0. 02032375197	0. 02839780279	0. 01733849503	0. 01564107568	-0. 01164351983
[3]Intercept	0. 02205769121	0. 01733849503	0. 0306984902	0. 01900405509	-0. 01332451448
[4]Intercept	0. 02503592854	0. 01564107568	0. 01900405509	0. 03496993268	-0. 01369350197
[1]POVERTY	-0. 03040566073	-0. 01164351983	-0. 01332451448	-0. 01369350197	0. 0268448611
[2]POVERTY	-0. 0130003275	-0. 01839198883	-0. 009921935577	-0. 00781334334	0. 009953387438
[3]POVERTY	-0. 01330396951	-0. 01061669024	-0. 02025922217	-0. 01034506774	0. 01073151324
[4]POVERTY	-0. 01382531211	-0. 009971054894	-0. 01148022048	-0. 02028955453	0. 008952925494
[1]FEMALE	-0. 009353890658	-0. 006080847462	-0. 004689495724	-0. 007482436174	0. 0001388734165
[2]FEMALE	-0. 005419911383	-0. 005399560595	-0. 004996475583	-0. 004635442285	0. 0008265588269
[3]FEMALE	-0. 006776076699	-0. 003380665193	-0. 005533155331	-0. 006034469884	0. 00198483699
[4]FEMALE	-0. 008685212792	-0. 002688618614	-0. 003815832277	-0. 008579340338	0. 004027492155
[1]NONWHITE	0. 002424410502	0. 002517521489	-0. 001760320001	0. 0009788401654	-0. 006827944925
[2]NONWHITE	0. 002104665176	0. 004619341276	0. 002468885134	0. 001668089177	-0. 003856029553
[3]NONWHITE	-0. 000439128437	0. 001882844366	0. 0002415334121	-0. 001173231148	-0. 003855709744
[4]NONWHITE	-0. 001451484263	0. 001471097294	-0. 000237177168	0. 0001173027671	-0. 001988748465
	[2]POVERTY	[3]POVERTY	[4]POVERTY	[1]FEMALE	[2]FEMALE
[1]Intercept	-0. 0130003275	-0. 01330396951	-0. 01382531211	-0. 009353890658	-0. 005419911383
[2]Intercept	-0. 01839198883	-0. 01061669024	-0. 009971054894	-0. 006080847462	-0. 005399560595
[3]Intercept	-0. 009921935577	-0. 02025922217	-0. 01148022048	-0. 004689495724	-0. 004996475583
[4]Intercept	-0. 00781334334	-0. 01034506774	-0. 02028955453	-0. 007482436174	-0. 004635442285
[1]POVERTY	0. 009953387438	0. 01073151324	0. 008952925494	0. 0001388734165	0. 0008265588269
[2]POVERTY	0. 01655461324	0. 008818291009	0. 007299463214	0. 001597377862	-0. 0008350481387
[3]POVERTY	0. 008818291009	0. 0174161752	0. 008673574153	-0. 0004444375661	-0. 0001044107673

[4]POVERTY	0.007299463214	0.008673574153	0.0155982033	0.001835491592	3.769563216e-005
[1]FEMALE	0.001597377862	-0.0004444375661	0.001835491592	0.01405001395	0.007789137756
[2]FEMALE	-0.0008350481387	-0.0001044107673	3.769563216e-005	0.007789137756	0.01086439924
[3]FEMALE	-9.417636938e-005	-0.0004044250007	0.0007489343056	0.00865528679	0.007693997968
[4]FEMALE	-0.0003899773081	-0.0008613855373	0.0004935314292	0.008378753999	0.007422423038
[1]NONWHITE	-0.004523009498	-0.0008003992597	-0.001307815604	0.0001762366154	-0.001692115988
[2]NONWHITE	-0.004832409759	-0.00244793101	-0.002226736184	0.001034118966	-0.001876088878
[3]NONWHITE	-0.004100815973	-0.003441706237	-0.0008797278031	0.002007183251	-0.0001818946385
[4]NONWHITE	-0.002343714947	-0.0007986077311	-0.002770336636	0.002260495195	-0.0009771037397

IVEware Multiple Imputation Regression, Wed Jan 09 17:13:54 2002

3

SASMOD WITH CATMOD

	[3]FEMALE	[4]FEMALE	[1]NONWHITE	[2]NONWHITE	[3]NONWHITE
[1]Intercept	-0.006776076699	-0.008685212792	0.002424410502	0.002104665176	-0.000439128437
[2]Intercept	-0.003380665193	-0.002688618614	0.002517521489	0.004619341276	0.001882844366
[3]Intercept	-0.005533155331	-0.003815832277	-0.001760320001	0.002468885134	0.0002415334121
[4]Intercept	-0.006034469884	-0.008579340338	0.0009788401654	0.001668089177	-0.001173231148
[1]POVERTY	0.00198483699	0.004027492155	-0.006827944925	-0.003856029553	-0.003855709744
[2]POVERTY	-9.417636938e-005	-0.0003899773081	-0.004523009498	-0.004832409759	-0.004100815973
[3]POVERTY	-0.0004044250007	-0.0008613855373	-0.0008003992597	-0.00244793101	-0.003441706237
[4]POVERTY	0.0007489343056	0.0004935314292	-0.001307815604	-0.002226736184	-0.0008797278031
[1]FEMALE	0.00865528679	0.008378753999	0.0001762366154	0.001034118966	0.002007183251
[2]FEMALE	0.007693997968	0.007422423038	-0.001692115988	-0.001876088878	-0.0001818946385
[3]FEMALE	0.0107915576	0.009002250965	-0.001014061215	0.0003011689205	0.0009797795243
[4]FEMALE	0.009002250965	0.01290691708	-0.003616642864	-0.001152072042	-0.0004982590243
[1]NONWHITE	-0.001014061215	-0.003616642864	0.02971795184	0.01589353252	0.01555398522
[2]NONWHITE	0.0003011689205	-0.001152072042	0.01589353252	0.03202540109	0.01786364406
[3]NONWHITE	0.0009797795243	-0.0004982590243	0.01555398522	0.01786364406	0.02078697344
[4]NONWHITE	0.0002496697165	0.0004191414802	0.01139883564	0.01369979951	0.009969969149
	[4]NONWHITE				
[1]Intercept	-0.001451484263				
[2]Intercept	0.001471097294				
[3]Intercept	-0.000237177168				
[4]Intercept	0.0001173027671				
[1]POVERTY	-0.001988748465				
[2]POVERTY	-0.002343714947				
[3]POVERTY	-0.0007986077311				
[4]POVERTY	-0.002770336636				
[1]FEMALE	0.002260495195				
[2]FEMALE	-0.0009771037397				
[3]FEMALE	0.0002496697165				
[4]FEMALE	0.0004191414802				
[1]NONWHITE	0.01139883564				
[2]NONWHITE	0.01369979951				
[3]NONWHITE	0.009969969149				
[4]NONWHITE	0.01737676649				

**References**

Gelman, A., Carlin, J., Stern, H., and Rubin, D. 1995. *Bayesian Data Analysis*. London. Chapman and Hall.

Heeringa, S.G., Little, R.J.A, and Raghunathan, T.E. 1997, "Imputation of Multivariate Data on Household Net Worth." University of Michigan, Ann Arbor, Michigan.

Kish, L and Frankel, M. 1974. "Inference from Complex Samples (with discussion)." *Journal of the Royal Statistical Society*, 36, 1-37.

Li, K.H, Raghunathan, T.E., and Rubin, D.B. 1991, "Large Sample Significant Levels from Multiply Imputed Data Using Moment-Based Statistic and an F Reference Distribution." *Journal of the American Statistical Association*, 86, 1065-1073

Raghunathan, T. E., and Rubin, D. B. 1988. "An Application of Bayesian Statistics Using Sampling/Importance Resampling to a Deceptively Simple Problem in Quality Control." *Data Quality Control: Theory and Pragmatics*, Editors: Liepins, G. E. and Uppuluri, V. R. R. New York: Marcel Dekker.

Raghunathan, T. E. (1994). "Monte-Carlo Methods for Exploring Sensitivity to Distributional Assumption in a Bayesian Analysis of a Series of 2 by 2 Tables." *Statistics in Medicine*, 13, 1525-153.

Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J., and Solenberger, P. 2001 "A Multivariate Technique for Multiply Imputing Missing Values Using a Sequence of Regression Models." *Survey Methodology*, Vol. 27, No 1.

Rubin, D. B. 1976. "Inference and Missing Data (with discussion)." *Biometrika*, 63, 581-592.

Rubin, D. B. 1987a. "The SIR-Algorithm- A Discussion of Tanner and Wong's " The Calculation of Posterior Distributions by Data Augmentation." *Journal of American Statistical Association*.

Rubin, D. B. 1987b. *Multiple Imputation for Nonresponse in Surveys*. New York. Wiley.



## Appendix

Example SAS data step for creating a pseudo STRATUM variable and a pseudo CLUSTER variable.

```
LIBNAME MYLIB 'C:\MYINDIR';  
DATA MYLIB.MYDATA;  
  SET MYLIB.MYDATA;  
  PSEUD_STRAT=1;  
  PSEUD_CLUST=_N_;  
RUN;
```