

A Manual for the Poverty and Inequality Mapper Module

REVISED VERSION: April, 2002

Gabriel Demombynes

*University of California-Berkeley, Department of Economics and
The World Bank, Development Research Group-Poverty Cluster*

1.0 PURPOSE OF MANUAL AND PROGRAM

The Poverty and Inequality Mapper Module is a SAS program that combines survey and census data to generate poverty and inequality estimates with standard errors at low levels of aggregation. The program generates such profiles for Atkinson inequality measures, Generalized Entropy class inequality indices, the Gini index, and Foster-Greer-Thorbecke (FGT) poverty measures. Use of the programs requires basic familiarity with SAS.

This manual provides a guide to the use of the Poverty and Inequality Mapper Module. It summarizes the module's purpose, identifies the user-defined inputs, describes how to prepare data for use with the program, discusses disk space requirements, and offers sources for further information and support.

2.0 DESCRIPTION OF PROGRAM

Poverty and inequality profiles at disaggregated levels have many applications. As a policy instrument, poverty and inequality profiles can be useful to target social spending. Additionally, as a research tool, such profiles can be employed to study the relationship between poverty/inequality and other variables, e.g. crime, health outcomes, or economic growth rates.

In the past, data shortcomings have made it impossible to generate detailed poverty and inequality profiles. On the one hand, surveys like the LSMS conducted by the World Bank in many developing countries have solid expenditure or income data, but at low levels of geographic aggregation they are not representative and lack sufficient sample size to construct poverty and inequality profiles. On the other hand, the national censuses carried out in many countries have sufficient population coverage, but do not include high quality information on expenditure or income.

The Poverty and Inequality Mapper Module employs a two-step method that overcomes these data shortcomings. In the first step, a series of regressions are run with the survey data. For these regressions, the left hand side (dependent) variable is the natural log of per capita expenditure in each household. The right hand variables are household demographic variables selected by the user. These variables must exist in both the survey and the census.

In the second step, the estimated parameters from the first step are applied to the census data to impute a value of log per capita expenditure for each census household. These imputed values are then used to produce poverty or inequality profiles for aggregated units of the census data.

The program is designed for general use and does not require the user to have an intimate understanding of the methodology's computational details. The user specifies a

survey dataset, a census dataset, and appropriate parameters. A strong feature of the programs is that they calculate not only point estimates but also standard errors.

The user is cautioned that the use of this module is the last step in a long process. The module can only be employed after extensive diagnostics and the variable selection process are completed. Moreover, this manual is not intended as a general guide to the poverty and inequality mapping methodology. Elbers, Lanjouw and Lanjouw (2001) provides additional information about the methodology in general. The following section offers only a summary version of the poverty mapping methodology.

3.0 METHODOLOGY

The basis of the approach is that per capita household expenditure for a household h in cluster c can be explained using a set of observable characteristics. These observable characteristics must be found as variables in both the survey and the census:

$$(1) \quad \ln y_{ch} = E[\ln y_{ch} | \mathbf{x}_{ch}] + u_{ch}.$$

Using a linear approximation to the conditional expectation, we model the household's logarithmic per capita expenditure as

$$(2) \quad \ln y_{ch} = \mathbf{x}'_{ch} \boldsymbol{\beta} + u_{ch}.$$

More explicitly, we model the disturbance term as

$$(3) \quad u_{ch} = \eta_c + \varepsilon_{ch}$$

where η_c is the cluster component and ε_{ch} is the household component. This error structure will allow both for spatial autocorrelation, i.e. a "location effect" for households in the same area, and for heteroskedasticity in the household component of the error. The two error components are independent of one another and uncorrelated with observable characteristics.

The model in (2) is estimated by Generalized Least Squares using the household survey data. The results from this first stage of the analysis are a set of estimated model parameters, including the beta vector, an associated variance-covariance matrix, and parameters describing the distribution of the disturbances.

In the second stage analysis we combine these parameter estimates based on the survey data with household characteristics from the census data to estimate welfare measures for subgroups of the census population. Specifically, we combine the estimated first stage parameters with the observable characteristics of each household in the census to generate predicted log expenditures and relevant disturbances. We simulate a value of expenditure for each household, \hat{y}_{ch} , based on both predicted log expenditure, $\mathbf{x}'_{ch} \tilde{\boldsymbol{\beta}}$, and the disturbance terms, $\tilde{\eta}_c$ and $\tilde{\varepsilon}_{ch}$ using bootstrap methods:

$$(4) \quad \hat{y}_{ch} = \exp(\mathbf{x}'_{ch} \tilde{\boldsymbol{\beta}} + \tilde{\eta}_c + \tilde{\varepsilon}_{ch}).$$

For each household, the two disturbance terms are drawn from distributions described by parameters estimated in the first stage. The beta coefficients, $\tilde{\beta}$, are drawn from the multivariate normal distribution described by the first stage beta estimates and their associated variance-covariance matrix. We then use the full set of simulated \hat{y}_{ch} values to calculate expected values of the average expenditure and the poverty headcount for the three spatial subgroups described above.

We repeat this procedure for z simulations, where z is typically specified as 100. For each simulation, we draw a new set of beta coefficients and disturbances. Then for each subgroup, we take the mean and standard deviation of the welfare measures and average expenditure over all z simulations. For any given location, these means constitute our point estimates, while the standard deviations are the standard errors of these estimates.

There are two principal sources of error in the welfare measure estimates produced by this method. The first component, referred to as *model error* in Elbers et al (2001), is due to the fact that the parameters from the first-stage model in equation (2) are estimated. The second component, described as *idiosyncratic error*, is associated with the disturbance term in the same model, which implies that households' actual expenditures deviate from their expected values. While population size in a location does not affect the *model error*, the *idiosyncratic error* increases as the number of households in a target population decreases.

4.0 USER-DEFINED PARAMETERS

The ALTMAP.SAS program itself is a text file which can be run as a SAS program. The SAS statistical package must be available in the operating system environment for the programs to function. This version of the program has been developed with SAS Version 8.1 for Windows and may not work properly with other versions of SAS. To customize the program for a particular use, the user must specify a series of input variables. The user specifies these input variables by modifying the SAS macro-variable definitions at the beginning of the program.

Most user-defined parameters are specified as SAS macro variable definitions. These are of the following form:

```
%LET MACRONAME=USERPARAMETER;
```

where `MACRONAME` is the program's *permanent* macro variable name, and `USERPARAMETER` is the user's particular choice of value for the macro variable. The user should modify `USERPARAMETER`, and leave `MACRONAME` unchanged. The `USERPARAMETER` should be specified in all capital letters.

The user-defined parameters are grouped by category: 1. Files and Datasets, 2. Variable Names, 3. Welfare Measures, 4. Simulation Parameters, 5. Census Trimming, and 6. Error Checks. The parameters are explained in detail below.

4.1 FILES AND DATASETS

The user specifications of the files and datasets differ somewhat from the general format for user-defined parameters described above.

4.1.1 Directory locations

The user must specify the directory where the survey and census datasets are located and the directory where the output dataset should be placed. These should be specified as follows:

```
LIBNAME SRVLOC 'SURVEYDIRECTORYNAME' ;
```

```
LIBNAME CENLOC 'CENSUSDIRECTORYNAME' ;
```

```
LIBNAME OUTLOC 'OUTDIRECTORYNAME' ;
```

The user should modify `SURVEYDIRECTORYNAME`, `CENSUSDIRECTORYNAME`, and `OUTDIRECTORYNAME`, leaving the single quotation marks. `SURVEYDIRECTORYNAME` should refer to the directory where the survey dataset is located. `CENSUSDIRECTORYNAME` should refer to the directory where the census dataset is located. `OUTDIRECTORYNAME` should refer to the directory where the user wishes the output datasets and files to be placed. The user should use directory naming conventions appropriate for the operating system in use. In Windows, a typical set of statements would be as follows:

```
LIBNAME SRVLOC 'C:\My Documents\DATA\survey data' ;
```

```
LIBNAME CENLOC 'C:\My Documents\DATA\census data' ;
```

```
LIBNAME OUTLOC 'C:\My Documents\DATA\output data' ;
```

Name of survey dataset and name of census dataset

The two datasets names must be in a SAS format appropriate to the operating system in use. Note that as per standard SAS syntax, these permanent datasets are referred to with two-part names. The first part of the name is the appropriate SAS library, followed by a period. The SAS library for the survey dataset is `SRVLOC`, and the library for the census datasets is `CENLOC`. The second part of the name is the name of the actual dataset. Thus, for example, if the survey dataset file is named `MYDATA`, it should be specified as `SRVLOC.MYDATA`.

4.1.2 Name of output dataset

This dataset will contain the final poverty or inequality estimates. This is also a two-part name. The first part is `OUTLOC`, and the second part is a name of the user's choosing. The default is `OUTLOC.MAPRES`.

4.1.3 Name of output file

This output file will contain various diagnostic information, first-stage results, and a printout of the final poverty or inequality estimates. It should be specified with the full directory path, in single quotes. In Windows, a typical specification might be `%LET OUTFILE='C:\My Documents\DATA\results.txt' ;`

Note that regardless of what name the user chooses, the output file will be an ASCII text file. The .txt extension is optional; it simply allows Windows to recognize it as a text file.

4.2 VARIABLE NAMES

It is important that these be specified in all capital letters.

4.2.1 LHS: Dependent variable in survey

The log of each household's per capita expenditure.

4.2.2 SRVWT: Weight variable in survey

The household's survey sampling weight.

4.2.3 CENWT: Weight variable in census

The number of persons in the household.

For either weight variable, the user may specify NONE to not use weights.

4.2.4 CLUSTER: Cluster variable

The variable identifying sampling clusters in the survey and the census. Using the survey data, the program will model a cluster effect at this level. With the census data, the program will draw a location component of the residual for each area defined by this variable. In the most common case, this variable will identify enumeration areas in the census, which correspond to the clusters used for the survey sampling. However, it is not strictly necessary that the two variables represent the same level of aggregation in both the survey and the census.

4.2.5 RHS: Right hand side variables for main regression

These variables must be named and defined identically in the two datasets. They should be presented in list form, separated by spaces. An intercept term is assumed and does not need to be included in the list. Under SAS 8.1 for Windows, variable names may be up to 30 characters in length.

4.2.6 ZVAR: Right hand side variables for the heteroskedasticity regression, if the HETERO=YES option is chosen below. An intercept term is assumed and does not need to be included in the list. Variables in this list may include explicit functions of variables, using SAS arithmetical functions. The module will generate these variables "on the fly" during program execution, so it is not necessary to create them in the survey and census datasets beforehand. The functions may also include the predicted value of log per capita expenditure from the ordinary least squares version of the first stage regression. This can be referenced as the variable `_YHAT`. For example, if EDUC, WATER, and LIGHT are variables found in both the census and survey datasets, the following would be one possible valid specification for the heteroskedasticity variables:

```
%LET ZVAR = WATER EDUC*WATER LIGHT*(EDUC**2) _YHAT*EDUC;
```

The above specification would call for WATER, the interaction of EDUC and WATER, the interaction of LIGHT and the square of EDUC, and the interaction of predicted log per capita expenditure with EDUC.

In the output file, the results from the heteroskedasticity regression are listed with right hand side variables `_ZVAR1`, `_ZVAR2`, etc. These are the ZVAR variables specified by the user, in the same order as the ZVAR list.

4.2.7 CENGEO: Level of aggregation for output poverty measures.

The variable specifying the level(s) of aggregation desired for the output poverty or inequality measure. These variables identify the census subgroups discussed in the methodology section. They are referred to in the program comments as the “census region variables” or the “census aggregation units.” There can be more than one census region variable if the user desires output for a variety of levels of aggregation. Note that the census region variables do not need to be geographic variables, and may be other categorical variables like ethnicity or profession of household head.

4.3 WELFARE MEASURES

4.3.1 INDICES: Poverty and/or inequality indices to be calculated

Three types of index are available: Atkinson inequality measures, Generalized Entropy class inequality indices, and FGT poverty measures. Enter a list of indices, specifying each index by ATK, GE, or FGT followed by an underscore (`_`) and an appropriate parameter. The parameter is commonly referred to as epsilon for Atkinson measures and alpha for GE and FGT measures.

Additionally, the module always produces estimates of the mean of y , regardless of the other indices specified by the user. The module can also calculate the mean of y^k , where y is the household consumption (or income) measure and k is a user-specified value. The following example will calculate mean of y , several of the most common measures, and also the mean of y^2 and y^3 .

```
%LET INDICES= FGT_0 FGT_1 FGT_2 GE_0 GE_0.5 GE_1 GE_1.5 GE_2 ATK_0
ATK_1 ATK_2 MEANY_2 MEANY_3;
```

4.3.2 GINI: Calculate Gini index as well?

The module can also calculate the Gini index, but it may require much more computation time. To calculate the Gini index in addition to other measures, specify `GINI= YES`.

4.3.3 POLAR: Calculate Gini, median, median share, and Wolfson polarization index?

If this option is selected, the module will calculate the Gini index, median consumption, the median share of consumption (the fraction of total consumption consumed by the poorer half of the population), and the Wolfson polarization index. These also require considerable additional computation time.

4.3.4 POVLIN: Poverty line, for FGT indices only.

This should be defined as a level (not log) figure, e.g. 400.

4.4 SIMULATION PARAMETERS

4.4.1 NUMSIM: Number of simulations for bootstrap

Based on limited experience, 100 simulations are suggested to get precisely estimated point estimates and standard errors.

4.4.2 HETERO: Choice of whether to model heteroskedasticity.

Enter YES for heteroskedasticity, which requires specification of a heteroskedasticity model above (ZVARS), or NO to assume homoskedasticity.

4.4.3 HHERR and LOCERR: Form disturbance terms

Several options are available for the form of the distribution of the household and cluster (location) components of the error term. These are the distributions from which the program draws randomly generated error terms for the census observations. Except in one case, the forms of the two distributions are unrelated.

For LOCERR, the form of the location component, the user may choose any of the following options:

- NORMAL: draw the location component of the residual from a normal distribution
- T: draw the location component of the residual from a Student's T distribution, with degrees of freedom specified by the LOCTFREE macro variable
- NONPARAMETRICA: draw the household component of the residual from the empirical household components of the residual estimated with the survey data, using simple one-stage draws
- NONPARAMETRICB: draw the household component of the residual from the empirical household components of the residual estimated with the survey data, using two-stage draws. If this option is specified for LOCERR, it must be specified for HHERR as well. See below for a description of the two-stage draws.
- NONE: do not model a location component of the error term

Likewise, for HHERR, the form of the household component, the user may choose any of the following options:

- NORMAL: draw the household component from a normal distribution
- T: draw the household component from a Student's T distribution, with degrees of freedom specified by the HHTFREE macro variable.
- NONPARAMETRICA: draw the household component of the residual from the empirical household components of the residual estimated with the survey data, using simple one-stage draws from the complete pool of survey household residuals.
- NONPARAMETRICB: draw the household component of the residual from the empirical household components of the residual estimated with the survey data, using two-stage draws. If this option is specified for HHERR, it must be specified

for LOCERR as well. The location components are drawn first, one for each cluster in the census data. Then household components for each observation in that cluster are drawn. The household components are drawn from a more limited pool than they are with the NONPARAMETRICA option. The pool consists of the survey household residuals corresponding to the census cluster's particular location residual draw.

4.4.4 SEED: Random number seed.

This is the seed SAS uses for its random number generator. If the seed is set not equal to zero, the program will produce identical results on repeated runs with the same specification. If it is set equal to zero, the program will produce a new set of random draws each time the program is run, and consequently the results will vary somewhat. The user may wish to run the program with several different seeds to examine how robust the results are to different random draws.

4.4.5 CENNUM: Number of observations to use from census

This is a useful feature for testing the program. If a numeric value is entered, the program uses a randomly selected subsample of the census data. To conduct analysis using all available census records, the user should specify ALL.

4.5 CENSUS TRIMMING

4.5.1 MAXIMPUTE and MINIMPUTE

The program will trim census observations with imputed values of the left hand side variable above a maximum and below a minimum value. For each parameter, the user may specify a numeric value (in logs), or NONE to not perform any trimming. Alternatively, the user may specify AUTO, in which case the program chooses the largest (or smallest) predicted value from the first-stage survey regression as the maximum (or minimum) acceptable imputed value.

4.5.2 EPSILONBOUND and ETABOUND

For each census observation the program draws random household (epsilon) and location (eta) components of the residual. If these draws are outside specified bounds, the program will draw new values until it draws values inside the specified bounds. The user can define those bounds with a numeric value, or NONE to not perform any trimming. Note that a numeric value is interpreted as a scaling of the largest residual, in absolute value, from the survey analysis. In other words, setting EPSILONBOUND=2 limits the household component of the error draw to 2 times the largest household component of the residual from the survey. Alternatively, the user may specify AUTO; setting EPSILONBOUND=AUTO has the same effect the same as setting EPSILONBOUND=1.25. The output file details how many times new draws were made in subsequent "rounds" of drawing values.

4.5.3 ALPHABOUND and BETABOUND

In the process of bootstrapping, the module draws random vectors of alphas and betas (the coefficients on the heteroskedasticity and main regressions, respectively) from

multinormal distributions described by the first stage point estimates and variance-covariance matrices. It may be desirable to exclude such draws when they are extreme.

The user can cause some vectors of alphas and/or betas to be trimmed by specifying a confidence interval. For example, specifying BETABOUND=0.99 will cause vectors of betas which fall outside the 99% confidence space for the betas to be dropped, and new sets of betas to be redrawn (the same is true for ALPHABOUND). The user may specify BETABOUND=NONE or ALPHABOUND=NONE to not exclude any beta or alpha draws.

4.6 OTHER

4.6.1 DOCHKs: Check datasets for variables and missing values?

Enter YES or NO. If YES is selected, the program does various checks to ensure the datasets are present with all the specified variables. If there is any problem, the program aborts and explains the problem in a message in the output file.

4.6.2 REJECTS: List all rejected error terms in output file?

Enter YES or NO. When the program generates random error draws for the census simulations, it rejects those out of the range of the corresponding first-stage error components. If this option is selected, the output file includes a list of the draws that were rejected.

4.6.3 TABMEANS: Produce tables of basic statistics for input variables?

Enter YES or NO. If this option is selected, the program produces tables with the mean, standard deviation, maximum, and minimum for each variable in the RHS and ZVAR lists. The program produces one table for the survey dataset and one for the census dataset.

4.6.4 COMPARE: Produce matrices with comparisons by census regions?

Enter YES or NO. Users may be interested in whether the two point estimates for given level of census aggregation are significantly different from one another. Because the estimates for different areas are produced using the same set of first stage survey parameter estimates, they have a covariance. Consequently, the standard errors of the two point estimates are not sufficient information for a hypothesis test of their difference.

If this option is selected, for each level of census aggregation the program produces a matrix with all the pairwise comparisons of point estimates. The matrix entry in cell (i,j) are the fraction of the simulations for which point estimate i is larger than point estimate j .

Here is an example of the output matrix, where the census aggregation variable is counties numbered 1-6. In this case the welfare measure calculated was the headcount rate, and 100 simulations were used. In this example, in 64% of the simulations, county 2's headcount estimate is larger than county 1's point estimate, while county 4's estimate is larger than county 3's estimate in only 3% of the simulations. Note that by definition,

no point estimate is greater than itself, so all diagonal elements are zero. Similarly, in all simulations where point estimate i is greater than point estimate j , point estimate j is NOT greater than point estimate i . Consequently, the sum of a given cell and its mirror image across the diagonal is always one.¹

COUNTY	_1	_2	_3	_4	_5	_6
1	0	0.36	0.83	0.99	0.06	0.05
2	0.64	0	0.94	1	0.11	0.09
3	0.17	0.06	0	0.97	0.03	0.01
4	0.01	0	0.03	0	0.01	0
5	0.94	0.89	0.97	0.99	0	0.49
6	0.95	0.91	0.99	1	0.51	0

These values can be used to conduct one-sided hypothesis tests. Consider the hypothesis that county 1's true headcount rate is larger than county 4's headcount rate. In 99% of simulations, this was true (4's estimate was larger than 1's estimate in only 1% of the simulations.) We can interpret $1-0.99=0.01$ as a p-value for the hypothesis. This means that at the 5% level we cannot reject the hypothesis. In other words, we can say with fair certainty that county 1 has a higher headcount than county 4.

The program prints these matrices to the output file and also saves them as SAS datasets, with intuitive names starting with "compare." The sample matrix shown above would appear as a dataset named `compare_fgt_0_county`. These datasets will appear in the output directory defined by `OUTLOC`.

4.6.5 SYMSIZ and WORKSIZ: Specifications for SAS matrix calculations.

These are parameters that determine how SAS internally allocates memory for the matrix calculations (using PROC IML) it does for part of the analysis. The default settings, using 50 M of RAM allocated to "symbolspace" and 50 M allocated to "workspace," should work for a computer with 128 M of RAM installed. These settings *may* be too high for a computer with less memory. If the program fails to run properly and produces a message in the log file indicating that there is not enough memory available, the user should try reducing these values.

5. PREPARING DATA FOR USE WITH THE PROGRAM

The immediate steps required for preparing data for use with the program are the following:

- 1) Rename all corresponding variables to be identical in the survey and census datasets.
- 2) In the survey dataset, create the left hand side variable. This should be the natural log of the household's per capita expenditure, where household per capita expenditure is household total expenditure divided by household size.

¹ This statement ignores the case where two point estimates are equal. While this is rare, it may occur if, for example, two counties have headcounts both equal to either 1 (all poor) or 0 (no poor).

3) Ensure that each of the right hand side variable names is no more than 30 characters (for SAS 8.1 for Windows. Other platforms may have more stringent requirements).

Once the user has specified the parameters in the appropriate text file, the program should be invoked at the command line, e.g. "SAS ALTMAP.SAS," or run as a batch file from the Windows environment (within Windows, right click on the module and select "Batch Submit"). It is suggested that the user not run the module from within the SAS application window. When the program is run within the SAS application window, and it encounters a problem in the user specification, SAS will immediately quit, without saving the log file. This can make it difficult to diagnose problems.

Some additional notes:

- The program will run most quickly if both the census and survey datasets have been cut down to only those variables used in the analysis, i.e. those specified by the user.
- Internal variable names for the programs all begin with an underscore (_) prefix. To avoid conflicts, the user should ensure that no user-defined variables begin with an underscore. The variable name NONE should also not be used in the datasets.

6. OUTPUT

The program produces two main forms of output. The first form of output is a SAS dataset, with the name and location specified by the user. The second is a text file with first stage results, diagnostic information and the results printed as SAS output.

The results includes a number of variables, with an observation for each census region requested by the user. The variables includes all census region variables (CENGEOS) requested by the user, the poverty/inequality point estimates, and associated standard errors. These variables have intuitive names, e.g. FGT_0, SEFGT_0, ATK_2, SEATK_2, etc. Note that because SAS does not allow a decimal point to appear in a variable name, an underscore replaces a decimal point in output variable names. For example, if a measure like GE_0.5 is in the list of user requests, the variables GE_0_5 and SEGE_0_5 will appear in the output dataset.

The output also includes the following variables:
 NUMHH – number of households in the census region
 NUMPERS – number of individuals in the census region (i.e. the sum of the weights)
 MEANY – estimated average per capita expenditure in the census region
 SEMEANY – standard error of AVGY estimate

The program also produces a SAS dataset named GLSRESULTS with the GLS estimation results, along with asterisks indicating significance at the 10% (*) or 5% (**) levels.

7. DISK SPACE REQUIREMENTS

At a minimum, the user needs to have available free disk space equivalent to roughly three times the space taken up by the census dataset. Unfortunately, the exact

disk space requirements of a particular application can only be determined by trial and error.

8. MODIFICATION AND SUPPORT

Users may contact the program author, Gabriel Demombynes, at gabriel@demog.berkeley.edu for assistance with issues not detailed in this manual. Users are free to modify the program as they choose, but modified versions should not be circulated to others without permission of the author.