STRATIFICATION OF PRIMARY SAMPLING UNITS FOR THE CURRENT POPULATION SURVEY USING COMPUTER INTENSIVE METHODS

Paul W. Ludington, U.S. Bureau of the Census
Washington, DC 20233

I. INTRODUCTION

An essential design feature of demographic surveys conducted by the U.S. Bureau of the Census is stratification of their Primary Sampling Units (PSUs). Stratification clusters PSUs into strata, from which a subset of (sample) PSUs is selected. Survey costs are reduced by interviewing only in sample areas. However, strata produced during stratification need to be "homogeneous," so survey estimates derived from sample areas will also accurately reflect non-sample areas. The degree of stratum homogeneity and the achieved reduction in survey costs both depend upon the capabilities of the PSU stratification system.

During both the 1980 and 1990 Redesigns of demographic surveys at the U.S. Bureau of the Census, PSU stratification was accomplished almost entirely by using the Friedman-Rubin ("F-R") stratification system [1], built around a core algorithm called the hill-climbing pass by Friedman and Rubin [2]. Consideration is now being given to the use of alternative stratification approaches in designing future surveys, due to (1) needs for increased survey reliability and reduced survey costs, (2) advances in search and optimization methods used in stratification algorithms, (3) the use of computer-intensive methods and knowledge-based processing for solving complex problems, and (4) favorable experience at the U.S. Bureau of the Census with a prototype stratification search program ("L"), described in this paper.

L, using a different search algorithm than that of F-R, was developed by the author during the 1990 Redesign stratification of Current Population Survey (CPS) PSUs. It originated to stratify CPS PSUs for Alaska, since the F-R algorithm could not account for the widely-varying interviewer travel costs among Alaska PSUs. L randomly generated over five million stratifications to minimize Alaska's expected interview costs. This initial version of L became the basis for the current L sub-routine RANDOM, used to randomly enhance the initial stratification for each interactive search session. L now also includes a MOVE sub-routine (to minimize an unconstained criterion) and a SCREEN sub-routine (to find constrained solutions). In limited tests on actual Redesign data, L has demonstrated its operational comparability to F-R. However, the purpose of this paper is to initiate, through discussion of the L algorithm, the development of future stratification search algorithms, with sophisticated "learning" capabilities, that exploit modern computational power and are flexible enough to adapt to diverse survey stratification needs.

L has been tested on Minnesota CPS 1990 Redesign data, and one L interactive search session (described in Section V) found 85 better stratifications than any found during all F-R 1990 Redesign search runs for Minnesota. L's interactive sessions are flexible, allowing backtracking and the use of user input during the search process. Relatively few L search sessions seem required to find an acceptable final stratification for a state. L is written in Base SAS (a product of SAS Institute Inc.) and has only 264 lines of code. It is flexible to modify and easy to use. A VAX minicomputer was used for all search sessions, system resources were shared with other users, and no special priority processing or storage requirements were necessary.

Section II overviews the CPS PSU stratification process. Section III describes search concepts that underlie stratification search algorithms. L's four-phase search algorithm is given in Section IV. Section V compares the results of the best L interactive search session for Minnesota to the best F-R Minnesota results from all of its 1990 CPS Redesign search runs. Section VI outlines research directions for developing advanced stratification algorithms and their software implementations.

II. STRATIFICATION OVERVIEW

The stratification of Primary Sampling Units for demographic surveys at the U.S. Bureau of the Census occurs primarily during decennial survey redesigns as a multi-survey, time-constrained application on the Friedman-Rubin stratification system. Redesigns begin by defining, stratifying, and selecting survey PSUs. Then the redesign focus shifts to within-PSU sampling.

The Current Population Survey has participated in recent decennial survey redesigns. CPS PSUs designated as self-representing (SR) are always in sample. However, CPS non-self-representing (NSR) PSUs are stratified, using the F-R multivariate clustering algorithm that searches for best (lowest-criterion) constrained stratifications. One sample CPS PSU is selected from each NSR stratum. CPS PSU stratification seeks to reduce CPS costs and the variances of CPS estimates, while optimizing interviewer workloads. The criterion to be minimized is between-PSU variance on each state's stratification variables (scaled to ensure their proportionate influence). The total population of each CPS NSR stratum is constrained, so a self-weighting sample can be used. Stratum workloads are constrained to provide acceptable interviewer workloads for CPS sample NSR PSUs, and the estimate of each state's level of unemployment is subject to a fixed reliability requirement.

The Current Population Survey has a state-level design. Its PSU stratification occurs one state at a time and much of its stratification data is state-level data. Each CPS PSU is one or more contiguous counties or minor civil divisions within a state, and some CPS stratification data is collected at the PSU level. Stratification data is used to compute criterion values, stratum sizes, sampling intervals, and workloads for all stratifications evaluated. Research is used to select stratification (criterion) variables that are correlated with key state CPS estimates and that are stable over time. Most data is for the current redesign period (1990), although some data pertains to the last redesign (1980). Housing units, population, and labor force participation rates are projected to the implementation year (1995).

The stratification process repeatedly clusters a state's NSR PSUs into strata, uses the criterion and constraints to evaluate each generated stratification, and selects the best state stratification. Since most states have too many potential stratifications to evaluate them all, search strategies are used to determine which subset of a state's potential stratifications to examine. Algorithms representing these search strategies are coded into program sub-routines. Searches are formed by linking together these program sub-routines. Each search evaluates a (randomly-initiated) subset of the state's potential stratifications. Within each search, processing is controlled by user-supplied search parameters and by information the program "learns" while evaluating stratifications. A number of searches are generally required for each state to calibrate search parameters (including the number of strata to form) and to ensure that the best constrained evaluated stratifications are close in criterion value.

To select the final stratification for a state, the single best (lowest-criterion) stratification satisfying state stratum-size constraints is identified from search output. Stratum workloads for this stratification are equal to the (rounded) ratios of pro-rated stratum housing units to the average sampling interval for the ten best constrained stratifications, plus any sample used to supplement insufficient workloads. If all of its stratum work-loads are acceptable, this best stratification is selected as the state's final stratification. Otherwise, one of the other top stratifications may be selected, or more searches may locate additional candidates. Final CPS stratifications for all states are transmitted to the PSU-selection phase of CPS Redesign, where a single sample PSU is selected with pps from each stratum of NSR PSUs for each state.

III. SEARCH CONCEPTS

Exhaustive evaluation (the only known method guaranteed to find optimal stratifications of a state's NSR PSUs) cannot evaluate the many potential stratifications for most states, so search is used to find acceptable stratifications. Search methods are classified as "weak," "application-specific," or "mixed" [3]. Weak search is generic, with no reference to a specific problem domain. Application-specific search uses knowledge about a specific problem to constrain the number of alternatives considered. Mixed search is a combination of generic and application-specific methods. Search methods can also be classified as "blind" or "heuristic." Blind search performs an orderly evaluation of all alternatives until a solution is found. Heuristic search uses an evaluation function (criterion) to direct search toward the most-promising alternatives. Hill-climbing (the primary search method for both F-R and L) is a weak (heuristic) search method. However, both programs apply stratum-size constraints (specific problem information) and therefore have "mixed" search algorithms.

A state's potential stratifications (for a fixed number of strata) can be represented by a search space that assigns one NSR PSU to each axis. Integer coordinates represent stratum assignments, and each search point with all-integer coordinates represents a unique stratification. Distinct points that switch labels for entire strata represent equivalent stratifications. Feasible search points satisfy all stratification constraints. If a search space has no feasible points, then a different search space with more strata needs to be used. The criterion and all stratification constraints are real-valued functions defined on the search space. A stratification search algorithm determines the appropriate search space for each state (one with feasible points and the fewest possible strata), evaluates a subset of this space during search runs or interactive sessions, and identifies a final stratification for each state.

An algorithm's search performance is determined by the quantity and "quality" of the search points it evaluates (including its balance between quantity and quality) and its use of randomization to redirect the search process. The quantity of points evaluated is determined by the available computational power, the design of the algorithm to exploit that computational power, the number of independent searches performed, and the size and criterion-topography of the search space. L exploits current computational power to evaluate a large number of search points during each interactive search session.

The "quality" of search points refers to their low criterion values, feasibility, and usefulness in generating high-quality "children." To increase search quality, effective algorithms "learn" useful information as they evaluate points, such as the PSU combinations that compose strata, PSU contributions to the criterion and sampling interval, stratum contributions to the criterion, stratum sizes, and changes in these items as successive stratifications are evaluated. Since a single search can evaluate hundreds of thousands of stratifications, a vast amount of useful search information is processed. Search algorithms vary in the amount of search information they can store and effectively use. Both F-R and L have elementary "learning" capabilities, storing "reference points" (previously-evaluated, high-quality stratifications) and evaluating a subset of points no greater in distance from them than '2' (where "distance" equals the minimum number of their PSUs requiring stratum reassignments to transform the points into equivalent stratifications). Evaluating nearby points exploits the "continuity" of the criterion and constraint functions over the search space (where "continuity" means simply that nearby points, having many function components in common, tend to have close function values). Thus, by evaluating points close to high-quality reference points, F-R and L expect to generate other high-quality points, the best of which are then selected as new reference points.

Randomly-generated points usually have low quality (since high-quality points are very scarce). However, random points are useful (1) as initial stratifications for independent searches, (2) for enhancing initial stratifications (as discussed in Section IV), (3) for sampling search space topography around reference points (particularly local minima), (4) when combining stratifications (within a single search) that optimize distinct characteristics, and (5) when combining the best stratifications from multiple independent searches.

IV. THE "L" SEARCH ALGORITHM

L is run through interactive sessions. Each session progresses through four general search phases. In Phase 1, stratification data (identical to that used by F-R) is accessed, stratification variables are scaled, and useful data sub-totals are computed. Also, users enter values for search parameters in Phase 1 of each interactive session. These parameters include the number of strata to form, the seed to use in randomly generating stratifications in the RANDOM sub-routine, the minimum criterion value and stratum size for stratifications output by RANDOM, and final minimum and maximum stratum size constraints. (Stratum sizes are measured, approximately, in terms of 1990 civilian non-institutional population of persons at least sixteen years of age.)

Phase 2 (a single call to the RANDOM sub-routine) randomly generates an initial stratification for each session and randomly transforms this stratification into one whose criterion value and minimum stratum size satisfy the user-supplied constraints entered in Phase 1. The criterion is between-PSU variance on a state's stratification variables. Placing a lower limit on minimum stratum size also tends to lower the maximum stratum size, as all strata converge toward the average stratum size. To form the initial stratification, RANDOM uses the user-supplied seed to generate one random number for each NSR PSU. Each random number is multiplied by the number of strata and rounded, to convert it to the PSU's initial stratum assignment. If this initial stratification satisfies the above constraints, it is output and RANDOM terminates. Otherwise, the initial stratification is used as a "reference point" for generating other stratifications, by randomly selecting one of its NSR PSUs and randomly changing its stratum assignment. Any derived stratification with a minimum stratum size no lower than that for the current reference point becomes a new reference point. The first reference point with acceptable criterion value and minimum stratum size is output and RANDOM terminates. If two million stratifications are generated without finding an acceptable one, then RANDOM terminates and the user must enter looser constraints. RANDOM can range widely over the entire search space to find acceptable constrained stratifications. By comparison, F-R randomly generates, but does not randomly enhance, the initial stratification for each of its search runs.

Phase 3 uses calls to the MOVE sub-routine to search for the stratification that minimizes the unconstrained criterion. The number of required calls varies over the search space, and calls can be grouped into interactive entries. (About 30 MOVE calls were required for Minnesota sessions, and each call usually ran in less than 30 seconds.) The single Phase 2 output stratification is the input for the first MOVE call, and the lowest-criterion stratification evaluated during each call is the input for the next call. Each input stratification is a reference point, used to generate all possible points with a stratum reassignment for a single PSU or a stratum-exchange for a pair of PSUs. By comparison, F-R evaluates only stratum reassignments for individual PSUs during a maximum of thirty criterion-minimization calls (called hill-climbing passes by Friedman and Rubin in [2]). Thus, each L Minnesota session evaluated more than four times as many stratifications as each F-R run, when minimizing Minnesota's unconstrained criterion.

Phase 4 uses calls to the SCREEN sub-routine to search for low-criterion stratifications that satisfy final constraints on maximum and minimum stratum size. The user chooses the number and content of interactive entries to process, where each interactive entry consists of a number of SCREEN calls. Processing proceeds from one interactive entry to the next unless the user (monitoring search progress on a terminal screen) decides to return to the end of an earlier interactive entry. The single Phase 3 output stratification is input to the first SCREEN call of the first interactive entry. Later calls input the ten lowest-criterion stratifications evaluated during the preceding call that satisfy that call's constraint on minimum stratum size. The user (based upon current search status) inputs a constraint on minimum stratum size at the start of each interactive entry. Each later call within each entry uses a revised constraint (computed by the program) equal to the average minimum stratum size for stratifications satisfying the last preceding call's constraint on minimum stratum size. Thus, within each interactive entry, constraints are gradually tightened. Input stratifications for all SCREEN calls are reference points, used to generate stratifications in the same manner as within the MOVE calls of Phase 3. By comparison, F-R evaluates only stratum-exchanges for PSU pairs, not stratum reassignments for individual PSUs, during its constraint-satisfaction phase (called the size-adjustment pass in [1]). Any Phase 4 stratifications that satisfy the final constraints on both maximum and minimum stratum size are output, when encountered, to a single file. At the end of Phase 4, this output file is sorted by criterion value, and duplicate stratifications (those output by more than one call) are removed. This output file of feasible stratifications (along with similar files from any previous state search sessions) is examined to select a final state stratification, using the procedure described in Section II.

V. SEARCH RESULTS

L has been tested on 1990 CPS Redesign stratification data for Minnesota. Minnesota was selected as a challenging search task for L, since approximately 100 F-R search runs (over about a two-week period) were required to select a final stratification for Minnesota during the 1990 CPS Redesign. The best L interactive search session to date (using the final number of strata and constraint values used by F-R) took part of one afternoon and found 85 different stratifications that were better (had lower criterion values) than any of the best stratifications found by F-R during all of its 1990 CPS Redesign search runs for Minnesota.

During this L interactive session, the RANDOM sub-routine generated 5196 stratifications, reduced the criterion value from 0.52425 to 0.41283 (below the constraint of 0.43), and increased the minimum stratum size from 78,619 to 124,264 (above the constraint of 124,000). The single RANDOM output stratification was input to the first MOVE call. Each MOVE call adjusted the input stratum assignment for either one or two NSR PSUs. The unconstrained stratification output by the final (twenty-eighth) MOVE call had a criterion value of 0.13750, and this stratification was input to the first SCREEN call.

The screening process was divided into eight interactive entries consisting of a total of thirty-seven SCREEN calls. Each stratification generated within each call was evaluated against a minimum stratum size constraint of 149,000 and a maximum stratum size constraint of 187,500. During the screening, 85 different stratifications that satisfied these constraints and had lower criterion values than the best F-R Redesign stratification were found. The best constrained L stratification had a between-PSU variance on the four (scaled) Minnesota stratification variables of 0.15146, 2.18% lower than the best F-R constrained value of 0.15483. The F-R criterion values increased rapidly, and the tenth best constrained L value of 0.15262 was 16.54% below the comparable F-R value of 0.18286. (Since a state's best stratification for stratum-size constraints may not satisfy the required workload range, it is important that criterion values be low for alternative stratifications.) The L statewide sample size was 834, a reduction of 3.47% from the F-R sample size of 864. Final Minnesota stratifications from both programs had satisfactory stratum workloads.

VI. RESEARCH AGENDA

To develop an "intelligent" stratification system for use by multiple surveys in diverse applications, a research program should (1) identify survey stratification objectives and applications, (2) design (modular) search algorithms in response to survey search requirements, and (3) develop flexible and efficient programs to implement the search algorithms.

Survey stratification objectives (involving cost reduction, survey reliability, and efficient operations) should be clearly specified, including acceptable tradeoffs. Stratification is used in both the design of entire surveys and the design of samples for testing alternatives to current survey methods. Stratification objectives and applications determine search requirements, since they determine search space dimensions, criterion complexity, and the number of constraints to be satisfied. Additional screening constraints could include stratum workloads, statewide sample size, expected interview cost, expected overlap with a survey's old design (by approximating overlap probability [4]), and expected impact upon the distribution of households by interview mode. Survey requirements could be included as multiple goals (rather than as fixed constraints), and cooperative game theory might be used to select among highly-qualified alternative stratifications.

Search-algorithm design matches search "demands" (from survey requirements) with search "supplies" (methods for finding acceptable solutions). Search algorithms should be based upon rigorous mathematical and statistical foundations, with precise definitions for the search space and its properties, search functions and their properties (especially concerning optima for nonlinear criteria and feasibility for multiple constraints), sample designs for selecting search points to evaluate, and search decision rules based upon conditional probabilities and expected values. Statistical pattern recognition (SPR) might yield fruitful insights. The discrete search space discussed in Section III may be extended to include points with non-integer coordinate values, for better representation of concepts of non-linear programming and use of surface-fitting approximations to the criterion.

Search algorithms are sequences of search steps of various types. Decision rules (explicit or implied by interactive entries) determine the nature and scope of each successive search step. Research could expand the variety of search steps and decision rules used to select them. Random generation of points (outlined in Section III) could be incorporated into additional search step designs. Decision rules for selecting reference points could include explicit measures of the expected "quality" of their children (where "children" refers to the set of all derived search points, no matter how great their distance from the reference point). Weights, computed by correlating PSU combinations in strata with their stratification criterion values, could help select both reference points and their children. Interactive decision-making could be assisted by creating graphic representations of the multidimensional search environment. Optimal control theory might assist in determining efficient "trajectories" through the search space for interactive search sessions.

Development of intelligent search programs requires the use of appropriate software (such as LISP or Prolog) for the effective (knowledge-based) processing of search information. Areas of computer science that deserve examination include Artificial Intelligence (AI), Expert Systems, and Neural Networks (both for neural learning and forecasting). Parallel processing could be used to assign multiple independent searches to distinct processors. Program sub-routines, encoding search algorithms, would be combined into modules related to specific survey search requirements. Program search performance and operational characteristics should be thoroughly tested (including the balance between automatic and interactive processing) across a wide range of geographic areas and constraints. Performance differences should be correlated with differences in algorithm structure and program implementation.

REFERENCES

[1] R. S. Jewett and D. Judkins. Multivariate stratification with size constraints, SIAM J. Sci. Stat. Comput., Vol. 9, No. 6 (1988), pp. 1091-1096.

[2] H. P. Friedman and J. Rubin. On some invariant criteria for grouping data, J. Amer. Statist. Assoc., 62 (1967), pp. 1159-1178.

[3] L. F. Pau and C. Gianotti. Economic and financial knowledge-based processing, Springer-Verlag, Berlin (1990), pp. 190-191.

[4] L. R. Ernst. Formulas for approximating overlap probability when maximizing overlap, Bureau of the Census, SRD Research Report Number: Census/SRD/TN-90/01 (1990).