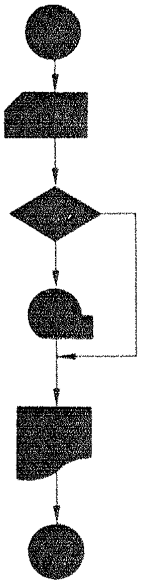
THE NORTHEASTERN FOREST-INVENTORY DATA-PROCESSING SYSTEM. VIII. DESCRIPTION OF SUBSYSTEM OUTPUT.



by
Robert W. Wilson Jr.
and Robert C. Peters

U. S. FOREST SERVICE RESEARCH PAPER NE-76
1967
NORTHEASTERN FOREST EXPERIMENT STATION, UPPER DARBY, PA.
FOREST SERVICE, U.S. DEPARTMENT OF AGRICULTURE
RICHARD D. LANE DIRECTOR

About the Authors

ROBERT W. WILSON, JR. took his Bachelor's degree at The Pennsylvania State University in 1947 and his Master's and Ph.D. degrees at Yale University in 1948 and 1965, respectively. He joined the U. S. Forest Service in 1948 and has worked in various research capacities for the Northeastern Forest Experiment Station. From 1961 to 1965 he was in charge of the Station's biometrics unit at New Haven, Conn. He is assigned at present to the Forest Insect and Disease Laboratory at West Haven, Conn.

ROBERT C. PETERS obtained his Bachelor's degree from the University of California in 1960 and his Master's at Yale University in 1961. He joined the Forest Service in 1961 as a research forester, and was assigned to the Station's biometrics unit from 1961 until 1965, when the unit was discontinued. Mr. Peters played a key role in the development of the data-processing system reported here.

THE NORTHEASTERN FOREST-INVENTORY DATA-PROCESSING SYSTEM. VIII. DESCRIPTION OF SUBSYSTEM OUTPUT.

Contents

PREFACE

THIS paper is the eighth in a series of ten papers prepared to describe the forest-inventory data-processing system of the Northeastern Forest Experiment Station. This system was devised for using modern, large-scale, high-speed computers in processing forest-inventory data. The series will comprise the following papers:

I. Introduction.

II. Description of subsystem EDIT.

III. Operation of subsystem EDIT.

IV. Information for programmers — subsystem EDIT.

V. Description of subsystem TABLE.

VI. Operation of subsystem TABLE.

VII. Information for programmers—subsystem TABLE.

VIII. Description of subsystem OUTPUT.

IX. Operation of subsystem OUTPUT.

X. Information for programmers —
subsystem OUTPUT.

VIII-A. INTRODUCTION

ONE of the major projects of the U. S. Forest Service is a nationwide forest survey, which is designed to obtain useful and timely information about the timber resources of the United States. In the course of the surveys, which are made mainly on a state-by-state basis, great masses of detailed data are collected about timber volumes, growth, timber cut, and other characteristics of the timber resource.

In recent years the volume of information obtained from forest-survey field plots has increased greatly. The task of compiling and analyzing this mass of data with mechanical computing machines was both cumbersome and time-consuming.

A solution to this problem was seen in the development of the high-speed electronic computers. The Northeastern Forest Experiment Station, which was responsible for conducting the forest survey of the heavily forested Northeastern States, investigated the possibilities and devised the Northeastern Forest-Inventory Data-Processing System.

This paper describes a part of the system, subsystem OUTPUT, which is a computer program designed to produce and print fully labelled tables of statistics for sampled populations from tabular summaries of the samples produced in program TABLE (see part V of this series). A complete understanding of that program is a necessary prerequisite to what follows.

The principal operations carried out by the program are the weighting and summing of selected input tables over all the samples in a population, the expansion of these sums to population totals, and the printing of the expanded tables, with labels that have been given as input. Several variations of this general procedure are provided to accommodate some of the more common sampling methods.

The program is written in the standard FORTRAN IV language, and is operative at the Yale University Computer Center on an IBM 7094/7040 Direct Coupled System under the IBSYS operating system with IBJOB processor.[1] It will operate with little or no modification on other comparable systems.

Part X in this series contains a selection of programming information that will be useful if the standard version of the program must be modified for any reason. Detailed instructions for setting up and executing jobs with the standard version are given in part IX. Copies of these publications and information on the FORTRAN IV source decks for the program can be obtained from the Northeastern Forest Experiment Station, 6816 Market Street, Upper Darby, Pennsylvania 19082.

VIII-B. PROGRAM OUTPUTS

The primary outputs from program OUTPUT are printed tables of statistics for sampled populations, selected from among the tabular summaries produced for the samples (data sets) in program TABLE.

The basic statistics put out for each selected table are cell-by-cell sums (including row and column totals) of the sampling-unit attributes contained in the table over all sampling units in the population, and the corresponding variances of these sums (except that the zero variances for populations wholly contained in the samples are not printed). In addition, tables of the corresponding standard errors of estimate, expressed either in absolute terms or as percentages, may be obtained on an optional basis.

Each statistic for each of the selected output tables is printed in the format specified for that table in the program TABLE job con-

[1] Mention of a particular product should not be construed as an endorsement by the Forest Service or the U. S. Department of Agriculture.

trol deck (see part VI-B sec. 320., in the description of that program). A maximum of 27 rows and 4 columns are printed per page, complete with table title, row and column headings (as supplied in the job control deck), and the name of the statistic. Larger tables are printed on multiple pages, each page being fully labelled.

Outputs equivalent to those just described can also be obtained on an optional basis for groups of populations. These statistics are simple sums of the individual population statistics.

In addition to, or in place of these primary outputs, some other modified outputs of the same general format can be obtained. For example, tables of population totals and their variances can be replaced by tables of means and their variances. Under certain conditions, the tables of population statistics can be replaced by tables of sample statistics; or, under an appropriate assumption of homogeneity within the population, by tables of statistics for particular segments of the population. However, modifications of this kind require judicious choice of a successful combination of input data, weights, expansion factors, and processing options. And they should not be undertaken without thorough understanding of the sampling method, the estimating problem, and the program logic and procedures.

VIII-C. DATA INPUTS

The data input to program OUTPUT must be a single magnetic tape file of binary records that contains the data set (sample) summaries for all populations that are to be processed in a given run. It may also contain summaries of data sets that are not to be processed in the run. These will simply be passed over during processing.

Thus, the magnetic tape file produced for these samples in program TABLE is the appropriate input to program OUTPUT, provided that care has been taken to reconcile the output option of the former with the processing option of the latter (see VIII-D). It is also necessary that the data set summaries applicable to a given population appear one after the other and in known order in the input file. Consequently, the same order should exist in the input file to program TABLE (see V-C in the description of that program).

The formats of the input file to program OUTPUT are fully described in part VII-F of the program TABLE description.

VIII-D. PROGRAM LOGIC AND PROCEDURES

The underlying logic and procedures of program OUTPUT are quite simple (fig. 1). The first step always is to read the control deck into the computer, interpret the specifications it contains, and store the necessary information. The interpretation includes the setting up of the processing option (see below) and the output options specified in the control deck. The control deck also contains the labels to be printed with the output tables, and all weights and expansion factors required in processing the job. In the general case, this setup phase is followed by several distinct phases in which the actual processing is accomplished.

First, the entire set of tabular summary data for a sample (data set) from a given population is read into the computer; and the selected tables are weighted and summed to population totals. This phase is repeated until all samples from the population have been processed.

Second, the final population variances are computed and, if elected, so are the standard errors of estimate. After all statistics are multiplied by the appropriate expansion factors for the estimate the tables of population statistics are printed. At this point, the tables of population statistics will be summed to totals for groups of populations, if that option has been specified. This phase is repeated until all populations specified for the job have been processed and printed.

In the final phase, the tables of statistics for groups of populations, if any, are printed and the program branches back to the beginning—the reading of the control deck for the next job. This phase is repeated until all jobs in the processing run have been completed, at which point the program is halted and the run is finished.

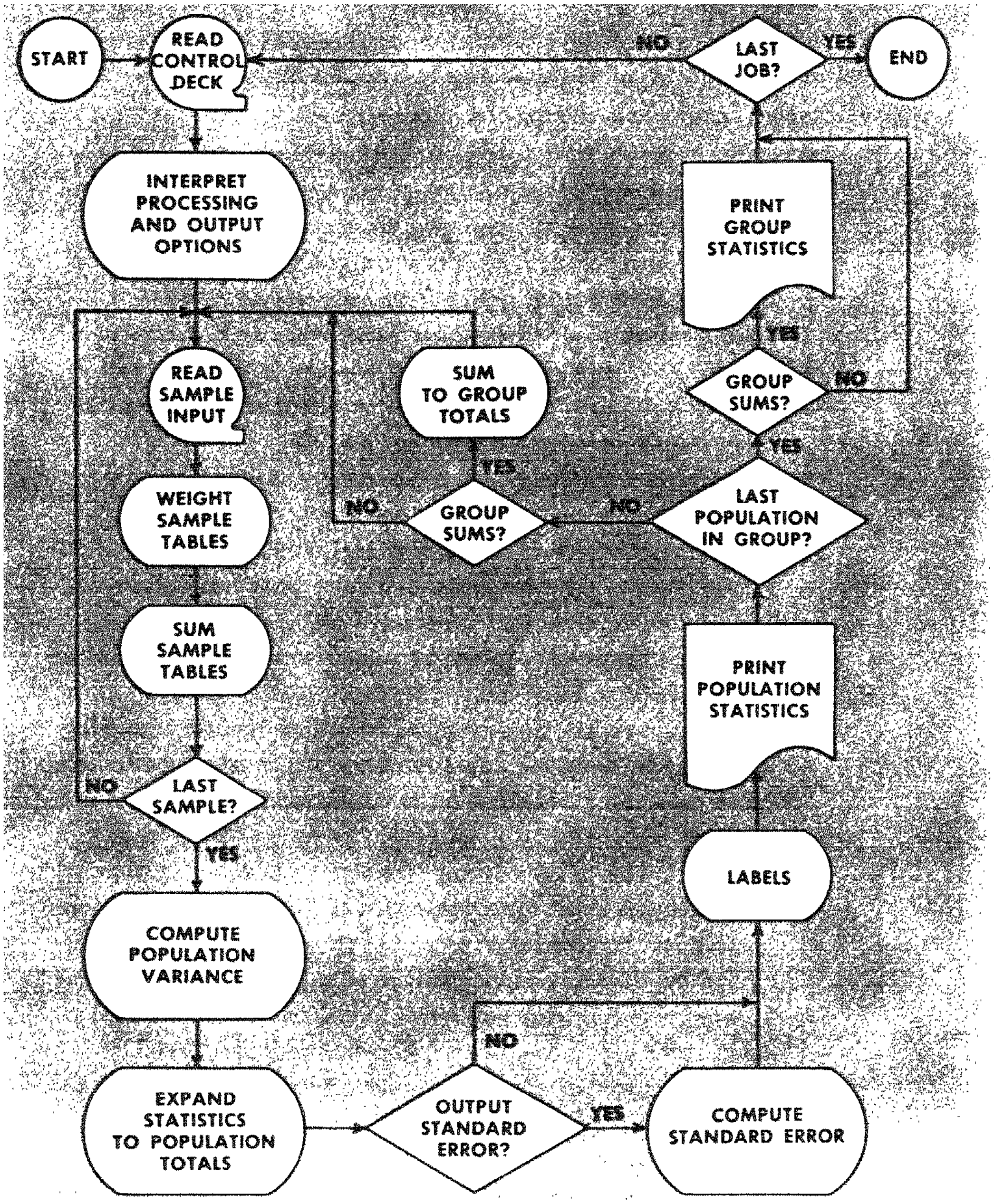


Figure 1. — A generalized flow chart of OUTPUT.

Actually, the program provides six processing options that are all special cases of the general case just described. The processing options consist of various combinations of the procedures contained in the first and second processing phases. Each option is described in detail in part X-D. In brief, they are:

1. Direct estimates of population parameters as simple sums of sample summaries; for example, when samples contain all elements of the population. These estimates require data input resulting from output option 1 of program TABLE.
2. Direct estimates of population parameters as expansions of summaries from a single sample; for example, a case of simple random sampling. These estimates require data input resulting from output option 3 of program TABLE.
3. Direct estimates of population parameters as expansions of the weighted sums of summaries from several samples, using known weights; for example, stratified random sampling from known strata. These estimates require data input resulting from output option 3 of program TABLE.
4. Direct estimates of population parameters as expansions of the weighted sums of summaries from several samples, using estimated weights; for example, stratified random sampling with double sampling for stratification. These estimates require data input resulting from output option 3 of program TABLE.
5. Ratio estimates of population parameters analogous to those of processing option 4, except that each cell of the tabular sample summaries is converted to a ratio of the tabular total and multiplied by an independent estimate of the tabular sample total before weighting and summing. These estimates require data input resulting from output option 4 of program TABLE.
6. Ratio estimates of population parameters analogous to those of processing option 4, except that each cell of the tabular weighted sums of the several samples is converted to a ratio of the tabular total and multiplied by an independent estimate of the tabular (population) total before computing the final variances and expanding the statistics. These estimates require data input resulting from output option 4 of program TABLE.

VIII-E. CONCLUSION

As described in the preceding chapters, the special purpose of program OUTPUT is to compute tables of population statistics from sample summaries, according to one of several different sampling schemes. The program is designed to be used in tandem with program TABLE so that both its inputs and outputs are controlled to a large extent by the outputs of that program; therefore, it is not well adapted for independent use in applications that do not also use program TABLE.

■