DOCUMENT RESUME

ED 211 335 SE 035 945

AUTHOR Novick, Melvin R.; And Others
TITLE Manual for the Computer-Assisted Data Analysis (CADA) Monitor (1980).
INSTITUTION Iowa Univ., Iowa City.
SPONS AGENCY National Science Foundation, Washington, D.C.
PUB DATE 80
GRANT NSF-SED-80-06357
NOTE 140p.; For related document, see SE 035 946.

EDRS PRICE MF01/PC06 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; *Data Analysis; Decision Making; Higher Education; Instructional Materials; *Mathematics; *Statistics; Undergraduate Study
IDENTIFIERS *Computer Assisted Data Analysis

ABSTRACT

The Computer-Assisted Data Analysis (CADA) Monitor is a conversational-language interactive computer software package intended for several audiences. It has three main functions: (1) to teach modern statistical methods to students with minimal mathematical backgrounds; (2) to provide educational administrators and others with easily used, yet sophisticated methods of exploring data sets, assessing utilities and prior probabilities, obtaining posterior probabilities, and combining these probabilities and utilities in order to produce coherent and effective decisions; and (3) to provide mathematically and statistically advanced persons with some powerful matrix tools useful in a wide variety of advanced applications. The system is accessible to students and educators who have no computer experience. The CADA leads a user through a statistical analysis on a step-by-step basis so that the user must do things correctly. A further innovation of CADA is the integrated tree-structure design of the system. Data are passed automatically from one module of an analysis to another. This manual discusses design concepts and capabilities of the Monitor as a data analysis system. (MP)

ED211335

Manual for The Computer-Assisted Data Analysis (CADA) Monitor (1980)

Prepared by

Melvin R. Novick, Robert M. Hamer,

David L. Libby, James J. Chen, and George G. Woodworth

The University of Iowa

Some of the materials incorporated in this work were developed with the financial support of the National Science Foundation Grant #SED8006357, Melvin R. Novick, principal investigator. However, any opinions, findings, conclusions, or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the Foundation.

035 945

MANUAL FOR THE COMPUTER-ASSISTED DATA ANALYSIS (CADA) MONITOR (1980)

PREPARED BY

Melvin R. Novick, Robert M. Hamer,

David L. Libby, James J. Chen, and George G. Woodworth

The University of Iowa

Some of the materials incorporated in this work were developed with the financial support of the National Science Foundation Grant #SED8006357, Melvin R. Novick, principal investigator. However, any opinions, findings, conclusions, or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the Foundation.

June 1, 1980

MANUAL FOR
THE COMPUTER-ASSISTED DATA ANALYSIS
(CADA) MONITOR (1980)

prepared by

Melvin R. Novick, Robert M. Hamer,

David L. Libby, James J. Chen, and George G. Woodworth

The University of Iowa

OVERVIEW

The Computer-Assisted Data Analysis (CADA) Monitor is a conversational-language interactive computer software package intended for several audiences. Leading a user step by step through a data analysis in much the same way as a computer-assisted instruction (CAI) program leads a student, CADA's first function is to teach modern statistical methods to students with minimal mathematical backgrounds. CADA's second function is to provide educational administrators and others with easily used, yet sophisticated methods of exploring data sets, assessing utilities and prior probabilities, obtaining posterior probabilities, and combining these probabilities and utilities in order to produce coherent and effective decisions. The third function of CADA is to provide mathematically and statistically advanced persons with some powerful matrix tools useful in a wide variety of advanced applications.

The purpose of this manual is to describe the 1980 release of the CADA Monitor. Earlier versions of the Monitor have been released beginning in 1973 (Novick 1973; Isaacs and Novick, 1976; Isaacs, DeKeyrel, and Novick, 1976; Novick, Isaacs, and DeKeyrel, 1977; Isaacs and Novick, 1978; Novick, Hamer, and Chen, 1979). This manual discusses the design concepts and resulting capabilities of the Monitor as a data analysis system for use on a wide range of computers. While this manual is not strictly necessary for the operation of CADA, a user may become more familiar with the Monitor by consulting it.

The primary innovative feature of the CADA system is the presentation of data analysis programs in the conversational mode (written in Basic), thus making the system accessible to students and educators who have no computer experience. CADA leads a user through a statistical analysis on a step-by-step basis so that the user must "do things correctly". The CADA system makes use of the two major strengths of the computer -- computational speed and memory -- as adjuncts to the human thought process and makes these two capabilities more directly usable than do other computing systems.

A further innovation of CADA is the integrated tree-structure design of the system. Data are passed automatically from one module of an analysis to another. This design facilitates a high level of freedom of movement within a given analysis, so that sections of an analysis can be redone without the need to start from the beginning. This is further facilitated by a sophisticated restart routine that gives the user a high degree of control over the analysis. Finally, the system is highly modularized so that new methods of analysis can be easily programmed using building blocks already available in the Monitor.

The 1978 version of CADA was installed at more than 100 universities and research organizations internationally. The 1980 version of the Monitor was released on June 1, 1980.

ACKNOWLEDGMENTS

The research reported here was supported in part by NSF grant SED 77-18432, Melvin R. Novick, Principal Investigator. The opinions expressed herein are solely those of the authors and do not necessarily reflect the opinions or policies of the supporting agency. The methods of utility-assessment underlying the assessment programs now used on the Monitor were developed with partial support from the Office of Naval Research under Contract N00014-77-C-0428.

We wish to acknowledge the considerable assistance of the following persons in the development of the Monitor back to the point when development began in 1971: Joseph Altmaier, Kirk Bayne, David Christ, David Chuang, Dennis DeKeyrel, Dattaprasad Divgi, Theresa Emerson, Les Finken, Gerald Isaacs, Charles Lewis, Roger Liddle, Shin-ichi Mayekawa, Victor Ormsby, Nancy Turner, Ming-mei Wang and Hiroshi Watanabe.

The translation of CADA-80 from an HP 2000 system to all other systems was designed and supervised by Raymond Novick. Assistance in translation was provided by Doug Brenner and Tim Bosserman. Arrangement for off-sites translations was facilitated by the cooperation of Ernest Hildebrand of the University of Missouri, Columbia and Thomas Moberg of Grinnel College.

In the earliest years of CADA strong encouragement, support, and guidance were provided by Gerald Weeg, late Director of The University of Iowa Computer Center and by James Whitely of the Center. In later years Howard Dockery, former director of the Weeg Computing Center, Lee Shope, assistant director, and Bobby Brown, director of the CAI Laboratory, lent support to this project. We are grateful to James Johnson, current director of the Weeg Center, for continuing support.

We would also like to acknowledge the assistance of Cher Wirth, Elsie Kuhn, Laura Peck and Julie Randall in the preparation of the manuscript. It should be noted that many of the statistical techniques for which analyses are available on the CADA Monitor are described and exemplified in the text, Statistical Methods for Educational Psychological Research, by Melvin R. Novick and Paul H. Jackson (McGraw-Hill, 1974).

Caution

The Computer-Assisted Data Analysis (CADA) Monitor (1980) was written and tested on an HP-2000 computer. It was then translated into other dialects of the Basic language for use on other computer systems. It is unfortunate, but computer programs are seldom error-free and translation undoubtedly introduces further problems. Extensive testing and error correction was performed for each dialect of CADA. We believe that CADA attains a high standard of accuracy. Because CADA is interactive and tree-structured, however, it is not possible to completely test CADA by following every possible path through the tree; it would take an unacceptable amount of time. Users should pretest those portions of CADA that are to be used as the basis for decision making.

We would appreciate being informed of any errors found. For this purpose a CADA Error Report Form is included at the back of this manual. Please include a hardcopy printout of the error, containing the entire sequence of events from the execution of CADA to the error. Additionally, a listing of the program as it is implemented on your machine would be helpful. We shall attempt to correct all errors that come to our attention and notify all members of the CADA Users Group of corrections that are relevant to their installations. Neither the developers of CADA nor The University of Iowa can accept any responsibility beyond that indicated above.

CONTENTS

I. INTRODUCTION

Many steps are involved in completing an informative statistical data analysis. Some are skilled tasks requiring the expertise of a substantive-area specialist; others are purely mechanical. The former include such tasks as choice of a model, specification of a prior distribution, specification of a loss or utility function, and interpretation of a posterior distribution. The latter include such things as the operations necessary to summarize and display data, to combine the prior distribution with data to produce the posterior distribution, and to produce probability statements from that distribution.

Unfortunately, it is all too often the case that arithmetic gets in the way of the professional's decision-making responsibilities by breaking the concentration and train of thought. At times, the sheer bulk of the computations precludes the use of advanced techniques by the unaided researcher. It is also true that good decision making requires the synthesis of many pieces of information at a critical decision point.

In order to facilitate the goal of informed data analysis for a wide range of users a monitoring system is needed that does all of the arithmetic, stores and then displays all necessary information at appropriate points, and, even further, guarantees that all of the steps in the analysis are performed correctly and in their proper sequence. This sequential guidance will be useful in teaching students by directing their steps through valid statistical analyses even if they do not yet understand fully what they are doing. It will also make it possible for investigators with minimal statistical backgrounds to perform complex statistical analyses with greatly reduced dependence on the tutelage of professional statisticians. Furthermore, good decision making can be facilitated by the use of computer memory to relieve demands on the investigator's memory.

Capitalizing on the capabilities of the computer to facilitate effective decision making, a system for computer-assisted data analysis (CADA) has been under continuing development at The University of Iowa since 1971 (Novick, 1973). Through ongoing investigation into available computer technology, coupled with expansion of the theoretical base on which the original system was constructed, the listing of available analyses has been continually expanded and individual algorithms refined. The organization of CADA through a Monitor system (Novick, 1973; Christ, 1973; Isaacs and Novick, 1976; Novick, Isaacs, and DeKeyrel, 1977; Isaacs and Novick, 1978; Novick, Hamer, and Chen, 1979) realized the stated goals of the project. The first major release of CADA was made in April 1976 under funding from the National Science Foundation.

Since CADA was meant for use both as an instructional tool and as a research tool for general application, it was necessary to find the most effective means of facilitating wide distribution of the Monitor for use on many computing systems. Due to limitations in available time, personnel, and money, reprogramming on a system-by-system basis was rejected as a feasible method of implementing CADA. Since no entirely transportable language existed for all interactive systems (including mini- and microsystems), it was decided to pursue a strategy that would permit interdialect translation rather than actual reprogramming. Examination of available hardware and sofware pointed toward the Basic programming language as the only alternative for translatability across several manufacturers. Isaacs (1972) demonstrated that programs written in one dialect of Basic could easily be translated into many other dialects, provided certain specified constraints were observed in the initial programming. The first Basic implementation of CADA was developed by Isaacs and Christ for the CDC 3600 at The University of Massachusetts in 1972 in the BASICX dialect. This was easily and quickly translated into the dialects for the Hewlett-Packard 2000 C and the Digital Equipment Corporation PDP-11, thus validating the assertions made by Isaacs (1972). Since then, the Monitor has been translated for use on many different machines ranging from microcomputers to the large, general purpose, timesharing machines. Earlier releases of CADA have been installed on the following systems:

Microcomputers

DEC PDP - 11V03 RT11 (1978 release)
WANG 2200 (1976 release)

Minicomputers

DEC PDP - 11 RSTS (1976, 1977, 1978 releases)
PDP - 11 RT11 (1978 release)
HP 2000 (1976, 1977, 1978 releases)

General Purpose

CDC CYBER NOS (1976, 1977, 1978 releases)
IBM 370 VS (1977, 1978 releases)
DEC PDP-10 TOPS-10 (1976 and 1977 releases)
UNIVAC 1110 (1977 release)

These machines have very different environments in which CADA must run. However, CADA has been designed to run effectively and efficiently in any of these environments. The 1980 release of CADA has been implemented on the HP 2000 (Basic), DEC PDP-11 (RSTS BASIC-PLUS), IBM 370 (VS VSBASIC), and PRIME 750 (PRIMOS BASIC).

II. THE CONVERSATIONAL USER-MACHINE INTERFACE

The way in which CADA has been designed to interact with the user is based on several assumptions that now, after several years of use in CADA, seem to have been validated. These assumptions are largely related to the way in which information is presented to and processed by the user. It is first assumed that data analysis will be facilitated if the user never has to rely on his or her memory. That is to say, at any given point in the analysis, the frame (simultaneous presentation of information) that the user sees must contain all information necessary for whatever decision must be made. Typically, these decisions are very simple ones; they may involve a quantitative evaluation or an indication of which branch the user wishes to take. In each case careful thought must be given to precisely what the user might want to know in making that decision.

Second, it was decided that the same system should be made available to all users regardless of their level of ability. This approach differs from that taken in other systems that have the capability of providing various levels of prompt for users with different levels of familiarity with the system. We have decided to use a single mode of presentation, having judged that it is difficult for even the user to know precisely what level of prompt may be needed at a given time. Thus, a user may be quite sophisticated but may have a lapse at a particular point and want additional information.

In order to make this system feasible without losing the interest of the experienced user by requiring the reading of redundant material, it has been necessary to put great emphasis on formatted and optional output. On the CADA Monitor the formatted output is done in such a way as to segregate information that is minimally sufficient for the experienced user from information that the less experienced user may need. Thus, the experienced user is able, because of CADA's carefully formatted design, to pick out what is needed by reading only a number or two, or perhaps a line or two, while ignoring blocks of text that will be important to other users. In many cases an explanation may be skipped by simply indicating it is not wanted. However, if the user at a later time wants the explanation, it is still available.

Our approach does involve the transfer of large amounts of information for each frame. Therefore, it becomes desirable that a minimum baud rate of 1200 (approximately 120 characters a second) be available so that the user will not be fatigued while using CADA. This makes the use of older hardcopy devices, such as the teletype, undesirable. This does still leave open the possibility of using dial-up facilities. While we sometimes use CADA with 300 baud output devices over phone lines, we find that users may become somewhat fatigued during lengthy sessions.

With the advances now being made in computer technology, relatively inexpensive equipment capable of operating at 1200 baud are now available. There are now several hardcopy devices on the market that operate at 1200 baud, including the GE Terminet and Decwriter Model III. Accoustic couplers are now available which operate at 1200 baud over telephone lines and facilities for their use are becoming increasingly available.

The third assumption made in the design of CADA is that it is important in the conversational user-machine interface for the expert to be able to enter an analysis at any point. For example, an expert, or indeed any user who knows precisely his or her prior distribution and therefore does not need help in this part of the analysis, could enter the analysis at the preposterior or posterior-distribution point.

Because the prior distribution module was not executed, the parameters of the prior distribution would not be available in the user's personal file. The Monitor is programmed to check for this, and if all necessary information is not available, the Monitor will ask the user to enter this information, which presumably the user can then supply. Also, the Monitor has the ability to let the user go back to any previous point in the analysis and redo the analysis from that point, reread the explanation, or review what has previously been displayed, while keeping all information and computations that were settled on in previous steps. In designing the package as we have, it has been our intent to give the user the feeling that the computer is working for the user, rather than that the user is working for the computer. The user should feel at ease working with the package.

The fourth assumption is that important user input should always be numerical. Yes-no switching is done by 1-0 coding. Judgments are entered numerically as scalars or vectors. The primary advantage of this approach is the ease of user input. A secondary advantage is that CADA can be run with very inexpensive user terminals (currently costing under $700), and indeed there should be little difficulty in using CADA with twelve key telephone-type keyboard as an input device and a modified TV set as display. A further detailed discussion of these properties of CADA is given in section IV.

III. THE FUNCTION OF CADA IN THE UNIVERSITY

One of the primary uses of the CADA Monitor is as a supplement for instruction in statistical methods courses taken by students in various disciplines. Students can learn most easily by doing, and the CADA Monitor facilitates this approach in the teaching of statistics. The Monitor also promotes the effective teaching of applied and mathematical statistics to statistics students.

Any program of instruction in statistics will involve the familiarization of the student with various standard statistical distributions, including normal, t, chi-square, and F distributions. Students can gain an understanding of these distributions by using the Monitor to compute characteristics and to display graphs of them. Students can do simple numerical exercises much more quickly by using the Monitor than by using the books of tables, and therefore students can do more such examples in a given period of time. It should be noted that the current version of the Monitor begins to make available the computation of various robust statistics in addition to the usual sufficient statistics under normal law assumption.

Further, the Monitor has the capability for the entry, editing, and storage of new data sets and provides some catalogued data sets for the student. These may be used in connection with set computational problems. The Monitor also has a capability for grouping, transforming, and regrouping observations so that fresh data sets can always be constructed. This is of great assistance to students performing exploratory data analyses in that it gives them a "feel" for the data.

A component group that instructs investigators in exploratory data analysis (EDA) is available. CADA now provides both bivariate and univariate EDA. The investigator is instructed on how an EDA analysis may be performed; the investigator is also provided with the necessary tools to look at univariate and bivariate data. Thus, CADA aids the investigator in developing a description of data.

Two further, related functions of the CADA Monitor within the university have been developed. Informal consultation between substantive workers and those associated with the CADA project have, in several instances, resulted in these workers doing more staisfactory, in-depth analyses of their data. The fact that the Monitor makes it possible for such persons to do this work with only minimal guidance from the consultant is beneficial in two respects. First, the consultant spends time only on those activities that make the best use of the consultant's skills. Second, the investigator works independently and thus ends up with a far greater "feel" for results and conclusions than if the work were done by others.

As a result of the experience gained to date with the consulting function of the Monitor, it is suggested that the Monitor could be used as a central tool in a university statistical consulting center. The method of operation would be very similar to that used in teaching the Bayesian statistics course with the Monitor. Substantive workers with statistical problems would consult with consulting-center staff members, who would help design the study, suggest appropriate statistical techniques, and get the investigator started in using the Monitor. The investigator would be responsible for preparing an initial write-up of the project, which would be reviewed by the consultant. The investigator would then do the analysis with the aid of the Monitor. After completing the analysis, the investigator would finish a draft of the paper, which would be reviewed by the staff member of the statistical consulting center. Following this initial review, the paper would be reviewed briefly by a more senior member of the center's staff to assure that the best possible advice had been given and that the more junior statistical consultant had benefited as much as possible from this particular internship experience. It is to be expected that, initially, appropriate methods would not be available on the Monitor for many of the problems facing the consulting center. However, the demands on the center would provide the primary guide for further development of the Monitor. This developmental work is currently in progress at The University of Iowa in collaboration with Professor George Woodworth, Director of the University Statistical Consulting Center.

A second possible function of CADA within The University would enable persons in mathematical statistics to see new statistical techniques that they develop made available for general use. Future versions of the CADA Monitor will contain new subroutines that will make it possible for mathematical statisticians to easily prepare their own programs for inclusion on the CADA Monitor. At present, some computational subroutines are available that may be used as building blocks for the construction of new modules. These subroutines provide the necessary computations to obtain cumulative distribution functions and highest density regions for various distributions. Also, matrix operators have been programmed for elementary matrix operations, for the standardization of a matrix, for sweeping of a matrix, and for the direct product of two matrices. It is our intention to add to this collection routines for performing one-dimensional numerical integrations, convolutions, eigenvalues and eigenvectors calculations, and a variety of other mathematical building blocks that will be useful in constructing complex statistical routines. With these available, it will be relatively easy for persons with only modest computer abilities in writing programs in Basic to construct conversational algorithms for inclusion in the CADA Monitor.

IV. DESIGN CONSIDERATIONS FOR THE MONITOR

The current Monitor was developed based on seven design considerations or rules. These are as follows:

1. The user is not required to have any programming skills. This was perhaps the most important consideration of all. Although possibly highly skilled in a particular subject area, the user may be quite unsophisticated in terms of computer skills. Therefore, user interaction must be kept as simple as possible. With CADA, the user need know only three system-related commands:

   1. how to sign on the system;
   2. how to start the Monitor running;
   3. how to sign off the system.

2. The Monitor must be self-documenting in terms of the options available. The Monitor should be modifiable to include new models, new techniques, and improvements in current programs, without the user having to search out the latest "newsletter" or update sheet. As noted in the Introduction, while this manual provides background information, it is not essential to the use of the Monitor.

3. The user should not be "left hanging" by the computer. If an invalid response is given by the user or a numerical integration fails to converge, an error message followed by a halt to the program is not adequate. The program must give the unsophisticated user enough information so that he or she can proceed. Furthermore, whenever possible, input from the user must be checked for validity to avoid system errors such as dividing by zero or taking the root of a negative number. In the CADA Monitor, when an unforeseen error occurs, the system chains to a routine in which the user is told to save the output for use by the person maintaining the system and then returns to the Monitor to continue the session.

4. The fourth design rule is "branchability", which enables the sophisticated user to determine a path through an analysis. In the CADA Monitor this is accomplished through the use of a four-level (component group, component, model, and module) tree structure and a "restart" facility. The user is allowed the option of terminating one part of the analysis at any point and choosing the point at which to "restart" the analysis. Whenever input is requested, the user may type "-9999" and enter the restart routine (RSTRT). Each user has a personal file that contains the data being used and all critical values computed for this user by CADA. An initiated restart permits the user to reenter the analysis at several earlier points with all necessary values retained in the individual's personal file.

5. The fifth central design strategy of the CADA Monitor is modularity. Whenever a statistical routine is constructed, the required steps are broken down into logical steps that are programmed separately as building blocks to be used in the present program and possibly in future programs. Thus, after nine years of development, an organized and catalogued collection of subroutines has been established, any member of which can be appended during program construction. The saving in programming time is obvious. However, a second benefit may not be so obvious.

Numerical analysis techniques have improved substantially during the past decade, and further significant improvements will be forthcoming. Because CADA is so highly modularized, any improvement in one building block subroutine automatically benefits all programs in which it is used, with little or no additional labor being required to modify these programs.

6. The sixth design strategy of CADA is a specific approach to readability that we believe is the correct one in this context, while not necessarily correct for other applications. First, we note that the CADA project is committed to lowering the cost of computation, making computation available to locations with minimal capital resources, and providing this capability to institutions having a wide variety of computer hardware, including those having only minicomputers or microprocessors.

These considerations rule out any possibility of providing an artificial intelligence-based dialogue with user-generated verbal input being syntactically or semantically parsed to provide instruction to the computer. Clearly, such an approach to statistical computation is desirable and attainable; moreover, we view our current work as being compatible with that approach. Any artificially intelligent reading of a user's input involving a semantic grammar will require a modularized response capability under the control of a monitor or mediator similar to that provided by CADA. At present, however, such an approach is not consistent with the cost-lowering goals that motivate this project. Large portions of the user's resources are necessary to support this type of user response.

A second common approach that we have not adopted involves making available two or three levels of textual accompaniment, depending upon a prespecified level of expertise of the user. This approach usually also incorporates a help call, which provides additional text. Our own trial use of these techniques convinced us of our inability to decide just how much text is necessary for each question for each of the levels of expertise. We also found that there is much variability in need for help among users. Our solution to the problem, therefore, is to present a full CRT screen of information when asking a question. This screen is highly formatted so that the expert can pick out the information needed without reading everything on the screen. This, of course, presupposes fast transfer of text at low cost, both of which can be achieved with minicomputers and microprocessors. With larger machines the text transfer can be expensive, depending on the charge schedule of the individual computer center. Negotiations with a cooperative computer-center administration at

The University of Iowa have been very helpful in this regard.

(With respect to charges, a brief parenthetical remark might be added. For CADA, the concept of economy of scale in selecting a computer is fallacious. For CADA, the larger the computer, the greater the cost. Use of a CDC CYBER might average out to an all-inclusive charge of $10 or more an hour for CADA, whereas use of an LSI-11 based microprocessor might be $1. Either figure, however, represents an incredible bargain. Users of CADA should discuss pricing policies with their computer center. The unusual mix of computer component usage in CADA should invite reconsideration of component charges.)

Regardless of what approach is taken to readability, we believe that the central concept should be reliance on the memory of the computer, not that of the user. As Sherlock Holmes pointed out, brain cells used to retain facts can not be used to process information. Indeed, the great virtue of the computer is that it can store facts and retrieve them instantaneously, thus freeing the mind to think. CADA is organized to emphasize fully the use of this strength of the computer.

7. The seventh design strategy for CADA involves transportability and translatability, which will be discussed in the next section.

V. TRANSPORTABILITY AND TRANSLATABILITY: SOME CONSIDERATIONS AND PROBLEMS

The most important development in the computer industry during recent years has been the rapid decrease in the cost of computers. This is true of central processors, memory, mass storage, peripherals, and terminals. The price/performance ratio of hardware has improved by a factor of 100 each decade since 1955, and indications are that this will continue at least for the next decade (Computerworld, August 8, 1977, p. 1). As a result of this trend, a purchaser of today's computer hardware can buy more than three times as much power for one-third the price of a computer five years ago.

While the cost of hardware has dropped dramatically, the cost of computer software has risen substantially over the past few years and will continue to do so in the future. This is especially true in the area of minicomputers and microprocessors, where new software is needed and must currently be amortized over a small base. With the increasing power and capabilities of minicomputers and microprocessors, there is a growing demand for new packages, which has not been met by software houses. In order that the gains in the hardware capability, particularly in microprocessors, be effectively translated into all-around increased computer effectiveness, the need for software packages must be met, and at a reasonable cost.

One means of counteracting the shortage and high cost of software is for users to pool and share their packages. Most hardware manufacturers have formed users groups, and in the educational context there is CONDUIT, based at The University of Iowa, whose purpose is to distribute computer software. In order that software be shared by many users, however, it must be transportable. This means that it must be written in a language that is supported on a wide variety of machines and that is compatible from machine to machine. Currently, the only language that fills these requirements is Basic. Unfortunately, the situation is complicated by the fact that Basic has many dialects that differ in both syntax and semantics. Different machines from the same manufacturer may not even use the same dialect of Basic.

The CADA monitor is currently developed on a Hewlett-Packard HP 2000 system in the Basic programming language; however, only a subset of the HP Basic is used. As shown in "Interdialect Translatability of the Basic Programming Language" (Isaacs, 1974) and "Basic Revisited" (Isaacs, 1976), programs written in one dialect of Basic can be easily translated to another dialect if a few simple rules are followed. Following these guidelines, translation of CADA from the HP 2000 dialect to several different dialects is straightforward. The 1977 release of CADA contained versions for use on the following computing systems: HP 2000 Basic, DEC PDP-11, DEC PDP-10, IBM 370(VS), CDC CYBER, UNIVAC 1100. The 1976 release had also contained a WANG 2200 version. The 1978 release had been made for the following systems: HP 2000 Basic, CDC CYBER (NOS, BASIC), DEC PDP 11V03 (RT11, MUBASIC), DEC PDP-11 (RSTS, BASIC-PLUS), and IBM 370 (VS, VSBASIC). The 1980 version of CADA has been released for the following systems: HP 2000 Basic, PRIME 450 and up (PRIMOS, BASIC and BASICV), DEC PDP-11 (RSTS, BASIC-PLUS), and IBM 370 (VS, VSBASIC).

Many statements that were used in the HP 2000 version translated directly across all dialects with no changes necessary. For example, GOTO, FOR, GOSUB, REM, and NEXT statements translated without change. Other statements were changed only if they had string or substring references. This modification varied from dialect to dialect but was accomplished under program control. Examples of this type of statement are LET, IF, PRINT, and DIM. Other statements such as GOTO-OF (ON-GOTO), and CHAIN, only needed a change in syntax from dialect to dialect. Statements involving formatted output and file input and output required extensive change, but were handled under program control across dialects. Therefore, only a limited set of options was allowed in the HP implementation of CADA.

The first step in planning for programs that could be translated from one system to another took place during the design phase. At that time it was decided what capabilities of the computer were necessary for CADA. It was decided that the following were necessary for a conversational, modularized package such as CADA:

1. Program chaining
2. External files
3. Retention of a personal file name across a chain
4. Formatted output
5. Limited string handling (substring, length, conversion)
6. Minimum of six digits of accuracy

The rules for translatability were then checked so that only the most translatable forms of statements were used. For example, only integer expressions were allowed to appear as a subscript of an array. This avoided the problem that some systems round the subscript while other systems truncate it.

The next step in planning for translation was to compile a list of statements used in the HP implementation of CADA that needed translation to the different dialects. These were kept to a minimum, since only a subset of the HP Basic statements were used. The following statements were modified for each translation:

1. PRINT #
2. READ #
3. PRINT USING
4. IMAGE (output format)
5. GO TO . . . OF
6. CONVERT
7. string dimensions
8. string functions
9. COM
10. CHAIN

A translator program was written for each dialect of Basic for which an implementation of CADA was to be produced. These programs resided on a PRIME 750 system. They modified the necessary lines of each module and produced an ASCII file that contained the translated programs. This ASCII file was then copied to magnetic tape on our PRIME 750. Each module of CADA was a separate file on the tape.

The translator programs handled about 99 percent of the statements, but some statements still had to be translated manually. The 1 percent that were not automatically translated were those that occurred very rarely and were easier to fix by hand than to make a special check for them in the translator. These occurred mostly in the control programs.

Data files to be transferred were written into ASCII files, one entry per record, and copied to a magnetic tape. These were then read by a small program written on the new machine and written out by the same program in the appropriate format for the new machine.

After translation, CADA was then thoroughly tested on the new machine to ensure that no errors had been introduced during the translation. This testing phase took several days. A tape of the modules and files was then created using the method that would facilitate the transfer of CADA to other machines of the same type and operating system. There were usually programs provided by the machine vendor which allowed easy transfer from one storage medium to another. These allowed all of CADA to be loaded from tape with a few simple commands. If a vendor-supplied program was not available, the easiest and most straightforward means was documented and sent to anyone receiving CADA.

It should be noted that there is an American National Standards Institute (ANSI) committee X3J2 that has been commissioned to standardize the Basic programming language. It has been meeting for over six years and has produced ANSI standard minimal Basic. This is a small nucleus from which extensions will be produced that will standardize enhancements. Minimal Basic contains only the following statements: DATA, DEF, DIM, END, FOR, GOTO, GOSUB, IF, INPUT, LET, NEXT, ON, OPTION, PRINT, RANDOMIZE, READ, REM, RESTORE, RETURN, and STOP. The following implementation-supplied functions were also standardized: ABS, ATN, COS, EXP, INT, LOG, RND, SGN, SIN, SQR, TAB, and TAN. More importantly, minimal Basic sets down syntax and semantics that are to be followed in future Basic dialects. The committee is now working on Level I enhancements to ANSI standard minimal Basic and plans to have these completed late in 1980. CADA has been developed in a manner consistent with the restrictions of the proposed Level I enhancements. When and if Level I Basic is standardized and adopted by all manufacturers, translation will no longer be necessary for CADA. But even an optimistic estimate would put that at least two or three years into the future.

Each dialect of Basic has it own semantics and syntax that limit what can be programmed in that dialect. When programming for translatability across several systems, the constraints become somewhat more demanding and at times preclude the use of features that may be present on one system only or that differ radically from one system to the next. This consideration, taken in conjunction with the seven design rules mentioned previously, has governed most of the design of the Monitor and its programs.

An important feature of the Monitor is that user input is screened for validity. If the user were allowed to enter responses as character input, a parser would have been necessary to allow for misspellings and synonyms. Also, since the string-handling capability is not highly developed in all Basic dialects and the handling of a finite set of responses can be done by much simpler coding, user responses to questions within the program segments have been forced to numeric form. Our decision to forego the extensive use of strings thus greatly enhanced the translatability of the system.

While the latest version of the Monitor was designed on a Hewlett-Packard 2000 system, an attempt has been made to minimize the dependence on features not available in most Basic dialects for other systems. As mentioned earlier, there are several features that have been used on the Monitor that may be somewhat limiting. These are:

1. Program chaining
2. External files
3. Retention of a personal file name across a chain
4. Formatted output.
5. Terminals

1. Program Chaining

Chaining is central to the logical design of the CADA system, since one of the cornerstones is that the user need know how to sign on but need not know the names of the individual modules. This necessitates either a main routine-subroutine system or a Monitor program that directs the loading and execution of the proper module. While the main routine-subroutine system would have had the advantage of ease of parameter passing, the number of parameters to be passed in our system is few, and the values can easily be stored in files and thus passed from one module to another. In addition, this structure is not translatable. The strategy of designing special master programs (CMONTR and CMPGRP) was therefore chosen, since this best suited most Basic programs.

The chaining strategy used in CADA has several advantages. First, only the module in use need be stored in the computer memory, thus reducing system overhead. Second, the system could be expanded with little effort on the part of a programmer and with no operational change visible to the user.

2. External Files

A file capability is needed to support the chaining strategy and to allow the restart feature. Since the user jumps from module to module in the course of an analysis, the CADA system must "know" with which analysis and module the user is working. This information is stored in the user's personal file along with any data that are needed in upcoming modules. Most Basic systems have a file capability; the Monitor uses a random access capability. This may be changed easily to sequential access by using different files for each of the random-access entry points (see section XIII).

3. Retention of a Personal File Name Across a Chain

The capability for multiple users of the system creates the need for a separate personal file for each active user. Thus the system always has to know which user is associated with which personal file. This presents a problem because a CHAIN destroys the values of all variables. On the HP-2000 system this problem is overcome by means of a COMMON statement, although other systems use different means. For example, this can be accomplished on a DEC PDP-11 by use of core common. Other systems use files that are temporary and unique to a terminal session. If there is no need for multiple users on the system, this capability need not be present.

4. Formatted Output

Formatted print statements are used to present information in a significant and visually-pleasing way. This is not necessary, per se, but is desirable, since the intended user is presumed to be not a computer expert, and therefore the user-machine interaction should be facilitated in every way possible. The formatted print statements and associated image statements do have analogues in every dialect to which we translate; however, there will be some change from machine to machine.

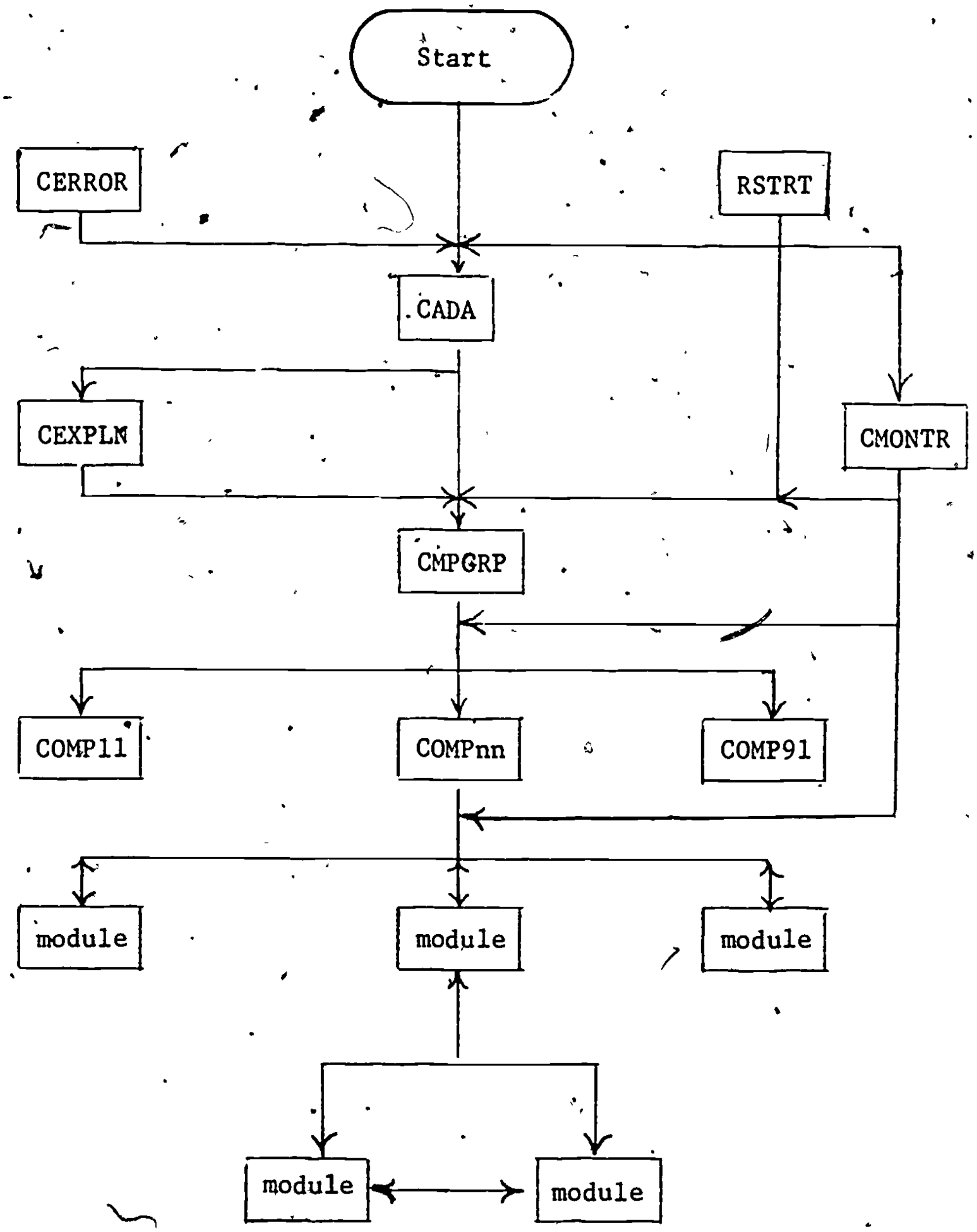
5. Terminals

The CADA Monitor can be used with a variety of teletype-compatible cathode-ray-tube terminals. To present information in a visually-pleasing manner, it is useful to erase the screen between frames when this is possible. Because each kind of terminal has a unique coding for the function of screen erase, it is necessary for the computer to ask the user to indicate which kind of terminal he or she is using. This is done with a multiple-choice-format question that includes an "other CRT" option. The appropriate code to be sent to the terminal is passed from module to module with the personal file name. The following terminals are now supported by CADA:

1. Beehive
2. ADDS or Hazeltine
3. Super Beehive
4. Hewlett-Packard 2640
5. Tektronix
6. Infoton and VISTAR
7. Lear-Siegler ADM-31
8. Imlac PDS 1G or Plato Plasma Tube
9. DECwriter III (LA120)
10. Other CRTs (a user specified code is used).
11. Other hardcopy terminals: Decwriter II, Texas Instruments Silent 700s, Terminets, teletypes, and all other hardcopy terminals

Figure 1

Flow Structure of the CADA Monitor



Note: All modules and COMPnn's may chain to CERROR, RSTRT, and CMONTR.

Supervisory Routines

There are six supervisory modules that monitor the presentation of the components to the user, plus a module for each component to control the presentation of the modules in the component. The following is a listing and brief explanation of the function of each routine.

1. CADA - Initializes the Monitor, chooses and initializes the user's personal file, sets the screen-erase characters for the user's terminal, and passes control to CEXPLN (if an explanation is desired) or to CMPGRP (if no explanation is desired).

2. CEXPLN - Gives a brief explanation of the structure and workings of CADA and passes control to CMPGRP.

3. CMPGRP - Controls the selection of the component group and the component within the component group.

4. CMONTR - Controls movement between modules and between levels of the tree structure, especially during a restart.

5. RSTRT - Called whenever the user enters "-9999" indicating that a restart is desired. RSTRT asks the user for the level at which to restart, sets the restart values in the personal file, and passes control to CMONTR.

6. CERROR - Called whenever a CADA Program encounters certain unforeseen errors, primarily when the personal file cannot be accessed.

7. COMPxx - Controls activity for component xx. Each component has a controller that monitors the selection of the model.

Explanation of Levels in the CADA System

There are four distinct levels in the structure of the Monitor: component group, component, model, and module. The highest level is called the component group level. There are nine component groups in the CADA structure at this time. A list of the component groups follows:

Component Group 1. Data Management Facility
Component Group 2. Simple Bayesian Parametric Models
Component Group 3. Decision Theoretic Models
Component Group 4. Bayesian Simultaneous Estimation
Component Group 5. Bayesian Full-Rank Analysis of Variance
Component Group 6. Bayesian Full-Rank Multivariate Analysis
Component Group 7. Elementary Classical Statistics
Component Group 8. Exploratory Data Analysis
Component Group 9. Probability Distributions

Each component group consists of a number of related component. This allows portions of CADA to be kept available on a fixed disk (on-line) or floppy disk while the remainder of the system is stored on a magnetic tape or other removable (off-line) medium. When one of these stored component groups is needed, it can be made available to the system by retrieval from the off-line medium. This allows on-line storage for CADA to be kept at a minimum. The combination of Component Group 9 with any other component group provides capabilities for both component groups. Component Group 9 can also be used alone. The modules and data files necessary for each component group are listed in section VIII.

The following is a listing of the component groups and the 24 components that are currently available with the models available under each component.

Component Group 1.	Data Management Facility
Component 12.	Data Movement (Input/Output, Editing)
Model 1.	Data Entry and Transfers
Model 2.	Data Display and Editing
Component 13.	Data Transformations
Model 1.	Nullary, Unary, and Binary Operations
Model 2.	Sufficient Statistics
Model 3.	Matrix Operations
Component 14.	File Maintenance
Model 3.	File Sorting (Data Grouping)
Component Group 2.	Simple Bayesian Parametric Models
Component 21.	Binary Models
Model 1.	Beta-Binomial Model
Model 2.	Beta-Pascal Model
Model 3.	Comparison of Two Proportions
Component 22.	Univariate Normal Models
Model 1.	Two-Parameter Normal (Natural Conjugate Priors)
Model 2.	Comparison of Two Normal Means
Model 3.	Comparison of Two Standard Deviations
Component 23.	Multicategory Models
Model 1.	Multinomial Model (Dirichlet Prior)
Component 24.	Simple Linear Regression Analysis
Model 1.	Simple Linear Regression Model
Component 25.	Multiple Linear Regression Analysis
Model 1.	Noninformative Priors
Model 2.	Informative priors
Component Group 3.	Decision Theoretic Models
Component 31.	Utilities and Expected Utilities
Model 1.	Assessment of Utilities
Model 2.	Evaluation of Utilities

Component 32.	Educational and Employment Selection
Model 1.	Quota-Free Selection (One Group)
Model 2.	Restricted Selection (Two Groups)
Component 33.	Selection of Educational Treatment
Model 1.	Assignment with Threshold Utilities
Model 2.	Assignment with Conditional Utilities
Component Group 4.	Bayesian Simultaneous Estimation
Component 41.	Simultaneous Estimation of Proportions
Model 1.	Arcsine Transformation
Component 42.	Simultaneous Estimation of Means
Model 1.	Equal Within-Group Variances
Component 43.	Simultaneous Prediction in m-Groups
Model 1.	Equal Slopes
Component Group 5.	Bayesian Full-Rank Analysis of Variance
Component 51.	Full-Rank Model I Factorial Analysis of Variance
Component 52.	Bayesian Analysis of Repeated-Measures Designs
Component Group 6.	Bayesian Full-Rank Multivariate Analysis
Component 61.	Bayesian Full-Rank Multivariate Analysis of Variance
Component Group 7.	Elemenatry Classical Statistics
Component 71.	Frequency Distributions
Model 1.	Absolute-Frequency Histograms
Model 2.	Contingency and Expectancy Tables
Component 72.	Summary Statistics
Model 1.	Summary Statistics
Component 73.	Graphic Displays
Model 1.	Absolute-Frequency Histograms
Model 2.	Bivariate Plots
Component 74.	Regression
Model 1.	Simple Multiple Linear Regression
Component Group 8.	Exploratory Data Analysis
Component 81.	Univariate Exploratory Data Analysis
Model 1.	Regular CRT Applications
Component 82.	Bivariate Exploratory Data Analysis
Model 1.	Regular CRT Applications
Component Group 9.	Probability Distributions
Component 91.	Evaluation of Probability Distributions

Each component and model is described in detail in section VI.

VI. MONITOR CONTENTS: DESCRIPTION OF MODULES WITHIN MODELS WITHIN COMPONENTS WITHIN COMPONENT GROUPS

COMPONENT GROUP 1 Data Management Facility

Component 11 (COMP11) Data Structures
(not yet implemented)

Component 12 (COMP12) Data Movement

This program controls the activity of Component 12.

Model 1 Data Entry and Transfers

This model allows data to be entered into the personal file from the terminal, or transferred between the personal file and certain disk files defined by the CADA system. There is a catalog of data sets, stored in disk files, that are accessible as well. The data set may be univariate or multivariate (maximum of 5 data elements), and it may be ungrouped or grouped (maximum of 12 groups). There may be a maximum of 1000 total values in the data set.

Module 1 (CMOD11) Data Entry from the Terminal

This module allows the user to enter small data sets from a terminal. The data are placed in the personal file where they are made available for further analysis. The data may be edited as they are entered.

Module 2 (CMOD31) Data Transfer from Disk

This module allows the user to transfer data from disk files (DFILxx) to the personal file. These data have been previously saved and given a data-set name by which the data can be identified. The disk files are part of the CADA system and are identified by passwords.

Module 3 (CMOD18) Data Transfer from the Catalog

This module allows the user to transfer data from one of the catalogued data sets to the personal file. The catalogued data sets are:

Data set #1: Iowa Tests of Basic Skills (ITBS)
School #1

These data, which are stored in disk file FILE1, are part of the data analyzed by Shigemasu (1976). The file contains 51 observations on five data elements (sex and ITBS scores on vocabulary, reading, arithmetic skills, and composite).

Data set #2: Iowa Tests of Basic Skills (ITBS)
School #14

These data, which are stored in disk file FILE2, are part of the data analyzed by Shigemasu (1976). The file contains 51 observations on five data elements (sex and ITBS scores on vocabulary, reading, arithmetic skills, and composite).

Data set #3: Emergency School Aid Act (ESAA)
Pilot Program Data

These data, which are stored in disk file FILE3, are part of the data analyzed by System Development Corporation (Wang, Novick, Isaacs, and Ozenne, 1977). There are five data elements (treatment or control, math pretest, math posttest, reading pretest, and reading posttest).

Data set #4: Iowa County Data

This set is data on the 99 counties in Iowa. There are four data elements: population, area, median income, and median number of school years completed (Isaacs, 1978). These data are stored on FILE4.

Data set #5: Sample Regression Data

These data are stored in disk file FILE5. Twenty-five observations from each of the ten schools of the 1968 junior college data (Novick, Jackson, Thayer, and Cole, 1972) were selected on the following data elements: English, mathematics, natural science, and GPA.

Data set #6: Sample ANOVA Data

These data were taken from Johnson and Leone (1964). There are 48 observations on 3 data elements (type of adhesive, curing pressure, and bonding strength).

Data set #7: Sample MANOVA Data

These data were taken from an experiment on communication (Bell, 1974). There are two between-subjects factors (substantive content and affective content of stimulus) and one within-subjects factor (round). Responses in the experiment were rated on three scales (substantive content, affective content, and meta-discussional content of response). The data set is in the form of summary statistics and thus is only appropriate as an example for the multivariate analysis of variance component.

Data set #8: Data from Junior Colleges

These data were taken from the 1968 and 1969 22-college data (Novick, Jackson, Thayer, and Cole, 1972). Data from 10 colleges are stored on files C8LL6, C8LL7, C8LL8, C8LL9, C8LL10, C8LL11, C8LL12, C8LL13, C8LL15, C8LL19 (1968 data), C9LL6, C9LL7, C9LL8, C9LL9, C9LL10, C9LL11, C9LL12, C9LL13, C9LL15, C9LL19 (1969 data). Each data set has five data elements: ACT scores (English, mathematics, social studies, natural science) and first-semester college grade-point average (GPA).

Module 4 (CMOD20) Catalog Data-Set Loading

This module loads the catalogued data set specified by the user into the personal file.

Module 5 (CMOD17) Data Transfer to Disk

This module saves the data in the user's personal file by transferring them to one of the files provided on disk (DFILxx). The user must provide the appropriate password and a data-set name which will be requested for identification at a later time.

Model 2 Data Display and Editing

This model allows the user to look at the data in the personal file. It also allows the user to modify individual values, to add data elements or observations, to delete data elements or observations, or to change the names of data elements.

Module 1 (CMOD12) Data Display and Editing

This module allows the user to display and edit the data in the personal file, add observations or data elements, delete observations or data elements, and change data element names. In the display, observations are presented in blocks of ten.

Component 13 (COMP13) Data Transformations

This program controls the activity of Component 13.

Model 1 Nullary, Unary, and Binary Operations

This model allows the user to modify values of a data set in a global sense. Most of the common operations and functions are available, such as addition, subtraction, multiplication, division, trigonometric functions (sine, cosine, tangent), logarithm and exponential functions, and various special functions (greatest integer, truncate, logodds, root-arcsine). In addition, several logical operations are available, such as "less than", "not equal to", and "or". Conditional sequences (IF-THEN-ELSE) are allowed, to a limited degree. Uniform and normal pseudo-random number generators are available.

Module 1 (CMOD15) Control

This module checks for the existence of a data set in the personal file and controls the flow of the model.

Module 2 (CMODE0) Explanation

This module provides a comprehensive explanation of the model.

Module 3 (CMODE1) Operation Definition

This module collects a set of operations and allows the user to edit them. Table 1 lists the set of available operations. The 'FILE' data elements are those that are, or are to be, in the data set stored in the personal file. The 'TEMP' data elements are those that are available for use during the transformation of a given observation.

TABLE 1

	OPERATIONS		**DATA ELEMENTS**	
NULLARY	UNARY	BINARY	FILE	TEMP
1. IF	101. X=NOT Y	201. X=Y AND Z	1. SEX	-1. UNUSED
2. THEN		202. X=Y OR Z	2. VOCAB	-2. UNUSED
3. ELSE	103. X=Y	203. X=Y EQ Z	3. READNG	-3. UNUSED
4. BEGIN	104. X=-Y	204. X=Y NE Z	4. ARTHSK	-4. UNUSED
5. END	105. X=1/Y	205. X=Y LT Z	5. COMPST	-5. UNUSED
	106. X=ABS(Y)	206. X=Y LE Z		-6. OBS #
	107. X=SIGN(Y)	207. X=Y GT Z		0. CONST.
	108. X=INT(Y)	208. X=Y GE Z		
	109. X=SQR(Y)			
	110. X=EXP(Y)	210. X=Y+Z		
11. X=UNIF(0,1)	111. X=LOG(Y)	211. X=Y-Z		
12. X=NORM(0,1)	112. X=LOG10(Y)	212. X=Y*Z		
	113. X=SIN(Y)	213. X=Y/Z		
	114. X=COS(Y)	214. X=Y^Z		
	115. X=TAN(Y)	215. X=LOGY(Z)		
	116. X=ATN(Y)	216. X=MAX(Y,Z)		
	117. X=LOGODDS(Y)	217. X=MIN(Y,Z)		
	118. X=ASINSQR(Y)			
	119. X=TRUNC(Y)			

ENTER AN OPERATION CODE (ENTER '0' TO TERMINATE).?

Module 4 (CMODE2) Operation Execution

This module executes the operations and replaces the old data set in the personal file with the new one. Data elements may be deleted or renamed.

Model 2 Sufficient Statistics

This model calculates various summary statistics from the data in the personal file.

Module 1 (CMOD13) Summary Statistics

This module provides summary statistics for the data in the user's personal file. It provides means, standard deviations, extremes, percentiles, midmeans, ranges, and correlation and covariance matrices.

Model 3 Matrix Operations

This model allows the user to manipulate matrices. Up to ten matrices may be stored, the maximum dimensions of which are ten by ten. Almost any statistical analysis that can be described as a series of matrix operations can be performed in this model.

Module 1 (CMODBO) Matrix Explanation

This module allows the user to obtain an explanation of the matrix package and to retrieve a matrix from the personal file.

Module 2 (CMODBP) Matrix Menu and Matrix Operations

This module offers the user a menu of matrix management, standard matrix operations, or special matrix operations. The module performs all matrix operations itself, and chains to CMODBQ for matrix management tasks. Standard matrix operations include matrix addition, subtraction, multiplication, transposition, calculation of X'X, inversion, and calculation of the determinant, as well as scalar addition, subtraction, multiplication, and division. Special matrix operations include standardization, direct product, and sweep operation.

Module 3 (CMODBQ) Matrix Management

This module performs matrix management functions (input, output, renaming, reshaping, and editing matrices).

Component 14 (COMP14) File Maintenance

This program controls the activity of Component 14.

Model 1 Directory Listings

(Not yet implemented)

Model 2 File Reorganization

(Not yet implemented)

Model 3 File Sorting

This model allows the user to group the data in the personal file into one to twelve groups. The grouping may be specified either by observation numbers or by the values of selected data elements. The groups may be chosen to be mutually exclusive or exhaustive. Any observations that are not selected are deleted from the data set.

Module 1 (CMOD16) Data Grouping -- set-up module

This module checks for data in the personal file. It also will ungroup the data if so desired.

Module 2 (CMOD61) Data Grouping -- working module

This module determines the method of grouping (observation numbers or data values) and sorts the data if necessary. The new, grouped data set replaces the old one in the personal file.

COMPONENT GROUP 2 Simple Bayesian Parametric Models

Component 21 (COMP21) Binary Models

This program controls the activity of Component 21.

Model 1 Beta-Binomial Model

This model confronts the problem of estimating the value of a proportion π. The prior distribution is assumed to be a beta distribution. The beta prior distribution along with the sample data forms a posterior distribution to be used in making inferences on the proportion π. It is assumed that π is from a binomial distribution. These modules make extensive use of subroutines BDTR (calculating probabilities) and BHDR (calculating HDRs).

Module 1 (CMOD2) Prior Distribution on Proportion

This module helps the user to develop a natural conjugate prior distribution on a proportion that is a beta distribution. Both a fractile assessment procedure (FASP) and a weighting procedure are used (Novick and Jackson, 1974). Coherence checks are provided along with the ability to respecify at different points. Means, modes, variances, percentiles, and graphs are displayed to help in the process.

Module 2 (CMODAB) Preposterior Distributions

This module helps the user to carry out an adversary preposterior analysis on a proportion. A rough idea of different sample sizes effect on the expected means of the adversary posterior distribution is given. Also provided is a closer look at the adversary distribution once a rough sample size has been determined (Jackson, Novick, and DeKeyrel, 1978).

Module 3 (CMOD3) Posterior Distribution

This module combines sample data with the prior distribution to form a posterior distribution. Means, modes, standard deviations, percentiles, and HDRs are provided for both the prior and the posterior distributions (Novick and Jackson, 1974).

Module 4 (CMODX) Predictive Distribution

This module allows the user to investigate the predictive distribution that is a beta-binomial distribution. For details see Component 91.

Module 5 (CMODB) Posterior Intervals

This module allows the user to investigate the beta posterior distribution that was termed in Module 3. For details see Component 91.

Model 2 Beta-Pascal Model

This model makes available standard methods for making inferences concerning a binomial proportion, assuming the data are available from a sequence of Bernoulli trials. The prior distribution is assumed to be the beta family and the sampling is Pascal. The beta distribution is evaluated using the subroutines BDTR (calculation of probability) and BHDR. See Applied Statistical Decision Theory, (Raiffa and Schlaifer, 1961) for details.

Module 1 (CMODAA) Prior Distribution on Proportions

This module assists a user to fit a beta distribution to his or her beliefs about a proportion. A fractile assessment procedure and weighting procedure are used to help assess his or her beliefs (Novick and Jackson, 1974). This module uses subroutines BDTR (calculating percentages) and BHDR (calculating highest density regions).

Module 2 (CMODAC) Posterior Distribution

This module combines the prior distribution with the sample data to form the posterior distribution that is a beta. Mean, mode, standard deviation, parameters, selected percentiles, and HDRs of the posterior and prior are displayed.

Module 3 (CMODY) Predictive Distribution (Beta-Pascal)

This module allows the investigator to examine the predictive distribution, which is a beta-Pascal distribution. (See Component 91 for details.)

Module 4 (CMODB) Posterior Intervals (Beta)

This module allows the investigator to examine further the posterior distribution, which is a beta distribution. (See Component 91 for details.)

Model 3 Comparison of Two Proportions

This model allows the investigator to examine the probability that the difference between two proportions is greater than some user-specified constant.

Module 1 (CMODT) Independent Beta-Distributed Proportions

This module allows the user to enter up to five values of the constant for which the probability that the difference is greater is calculated. This module makes use of the subroutine BDTR and the approximation given in Novick and Jackson, 1974, section 10-5.

Component 22 (COMP22) Univariate Normal Models

This program controls the activity of Component 22.

Model 1 Two-Parameter Normal (Natural Conjugate Priors)

This model confronts the problem of estimating the two parameters of a normal distribution, which are the mean (μ) and standard deviation (σ). Prior distributions on μ and σ are quantified. Two different types of preposterior analyses may be performed, one being an adversary preposterior, and the other a consensus preposterior. Sample data are combined with the prior distributions and posterior distributions on μ and σ are formed and may be investigated. This model is discussed in chapter 7, Novick and Jackson, 1974.

Module 1 (CMOD4) Prior Distribution on Standard Deviation

This module helps the user to quantify prior information on the standard deviation (σ) from a normal distribution. Both a fractile assessment procedure (FASP) and weighted procedure are provided. Coherence checks and the ability to respecify portions of the analysis are given. The prior distribution on σ is an inverse chi distribution (Novick and Jackson, 1974). Means, modes, standard deviations, percentiles, and HDRs are provided using CSQDTR (probabilities) and ICHDR (HDRs).

Module 2 (CMOD5) Prior Distribution on Mean

This module helps the investigator to quantify the conditional prior distribution on the mean (μ) given the value of σ. The value for σ is either passed from Module 1 or entered by the user. Both a fractile assessment procedure (FASP) and a weighted procedure are provided. Coherence checks and the ability to respecify portions of the analysis are given. The conditional prior on μ is a normal distribution (Novick and Jackson, 1974). Mean, standard deviations, percentiles and HDRs of this distribution are provided using the subroutine NDTR.

Module 3 (CMOD64) Adversary Preposterior

This module permits the investigator to obtain a prior estimate of what an adversary's posterior beliefs would be following a sample of prespecified size (Jackson, Novick, and DeKeyrel, 1977).

Module 4 (CMOD22) Consensus Preposterior

This module can be used before an experiment is performed to predict an adversary's reaction to the data that the experiment will yield. The shortest consensus interval, which is an estimate of the interval containing the population mean on which the decision makers could agree (Woodworth, 1976), is computed.

Module 5 (CMOD6) Posterior Distributions on Mu and Sigma

This module asks the user to enter the sample data: sample size, mean, and standard deviation. The sample data are then combined with the priors on μ and σ to form posterior distributions. The posterior distribution on σ is an inverse chi distribution with degrees of freedom ν and scale parameter λ^2. The posterior distribution on μ is a Student's t distribution with degrees of freedom ν, mean ξ, and scale parameter κ.

Module 6 (CMODC) Posterior Intervals for Mu

This module allows the user to examine the posterior distribution on the mean (μ). This distribution is a Student's t with degrees of freedom ν, mean ξ, and scale parameter κ. The user may examine the following:

1. Percentiles
2. Highest density regions
3. Probability less than (greater than) some value
4. Probability between two values
5. Graph

Module 7 (CMODD) Posterior Intervals for Sigma

This module allows the user to examine the posterior distribution on the standard deviation (σ). This distribution is an inverse chi with degrees of freedom ν and scale parameter λ^2. The user may examine the following:

1. Percentiles
2. Highest density regions
3. Probability less than (greater than) some value
4. Probability between two values
5. Graph

Model 2 Comparison of Two Normal Means

This model provides for the comparison of means from two univariate normal populations. It is assumed in this model that the variances are unknown and unequal. The two posterior Student's t distributions are assumed to be independent. A Behrens-Fisher comparison is made, (Novick and Jackson, 1974, section 8-2). The two posterior distributions may be obtained by performing two runs of Model 1 -- Two-Parameter Normal Model of Component 22.

Module 1 (CMOD7) Explanation of Two Independent t Distributions

This module explains the posterior Behrens-Fisher distribution on the difference of two normal means.

Module 2 (CMODF) Evaluation of Behrens-Fisher

This module allows the user to examine a Behrens-Fisher distribution with the same options as in Component 91. The user may view percentiles, HDRs, probabilities less than some value, probabilities in an interval and graphs of this Behrens-Fisher.

Model 3 Comparison of Two Standard Deviations

This model provides the comparison of two standard deviations from univariate normal populations, assuming that each has a posterior inverse chi distribution.

Module 1 (CMODU) Independent Inverse Chi-Distributed Standard Deviation

This module allows the investigator to examine the ratio of the two standard deviations. The probability that this ratio is less than some specified value is calculated by considering the F distribution of the ratio of the posterior distribution of the variances (Novick and Jackson, 1974, section 8-2).

Component 23 (COMP23) Multicategory Models

This program controls the activity of Component 23.

Model 1 Multinomial Model (Dirichlet Prior)

This model makes available the standard natural-conjugate analysis for the multinomial model with a Dirichlet prior distribution. Both prior and posterior distributions are Dirichlet distributions, where the parameters indicate the proportion of the population falling into each category. This analysis is described in chapter 10, section 7 of Novick and Jackson, 1974.

Module 1 (CMODG) Prior Distribution

This module allows the user to assess a joint prior Dirichlet distribution. The fact that the marginal distribution on the proportion in any one category is a beta distribution is used to help the user check coherence. The sum of the parameters of the marginal beta distribution for any category must equal the sum of the parameters of the joint Dirichlet distribution. The user is first asked for point estimations of the proportion in each category under the constraint that the estimates must sum to 1.0. These estimates are taken to be measures of the central tendency of the user's prior marginal distributions on the proportion in the respective categories. The user is then asked for the 25th and 75th percentiles of his or her prior marginal distribution on the proportion in each category. In general, the beta distributions fitted to the user's percentile specifications will not satisfy the coherency condition discussed above. It is therefore necessary to fit a set of beta distributions and a Dirichlet distribution that satisfy the coherency conditions. The 25th, 50th, and 75th percentiles of the fitted beta marginal distributions are displayed along with the joint estimates and the implied hypothetical sample size. The hypothetical sample size is equal to the sum of the parameters of the beta marginal distribution on the proportion in any category and, by coherency, to the sum of the parameters of the

Dirichlet joint distribution. The user can then change the hypothetical sample size and/or point estimates until he or she finds a satisfactory fit. Once he or she has indicated that he or she is satisfied, the parameters, mean, and mode of the Dirichlet prior distribution, and selected percentiles of the beta marginal distribution are displayed.

Module 2 (CMODH) Posterior Distribution

This module combines sample data with the user's prior distribution and forms the posterior distribution, which is a Dirichlet distribution. The parameters (A and B), mean, mode, and selected percentiles of the posterior are displayed.

Component 24 (COMP24) Simple Linear Regression Analysis

This program controls the activity of Component 24.

Model 1 Simple Linear Regression Model (Informative prior)

This model makes available a Bayesian linear regression analysis with one predictor (X) and one criterion (Y) variable. The user incorporates prior information with the sample data to form posterior distributions on the regression parameters. These may then be more fully examined. This procedure is discussed in section 9-1 of Novick and Jackson (1974).

Module 1 (CMOD65) Prior Distribution

This module helps the user to form prior distributions on the parameters: α, β, and σ. The first step is to fit a line to the best estimates of Y given X for different values of X. This model assumes that the uncertainty about a randomly sampled Y given X has two components. First, there is uncertainty as to what the expectation of Y given X is, and, second, there is the uncertainty (the residual uncertainty) that would be there even if the expectation of Y given X were known.

The 50th percentiles of the prior distributions on Y given X and on the expectation of Y given X are assumed to be equal. However, the 75th percentile of the prior distribution on Y given X is assumed to be greater than the 75th of the prior distribution on the expectation of Y given X, since the uncertainty about Y includes the uncertainty about the expectation of Y.

The prior on the residual standard deviation (σ) is an inverse chi distribution, and the prior on the slope (β) and intercept (α) are t distributions.

Module 2 (CMOD66) Posterior Distributions on Residual Standard Deviation and Slope

This module asks the user to enter the sample data: sample size (n), mean of predictor (X.), variance of X, mean of criterion (Y.), variance Y, and correlation of X and Y. These are combined with the priors to form posteriors on the residual standard deviation (inverse chi) and the slope (Student's t).

Module 3 (CMOD67). Evaluation of Posterior Distributions

This module allows the user to examine the posterior distributions on Y and the mean of Y. The user may look at percentiles, highest density regions, and probabilities less than specified values for X.

Module 4 (CMODC9) Save the parameters of the regression equation

This module permits the user to save the parameters of the regression equation in the personal file to be used by Component Group 3 for decision analysis.

Component 25 (COMP25) Multiple Linear Regression Analysis

This program controls the activity of Component 25.

Model 1 Noninformative Priors

This model provides a standard Bayesian analysis for simple or multiple regression model assuming noninformative prior distributions. The analyses are discussed in Bayesian Inference in Statistical analysis (Box and Tiao, 1973).

Module 1 (CMODC5) Selection of the independent and dependent variables to form the regression model.

This module displays the variables in the sample data and requests the user to select the independent and dependent variables for the regression model.

Module 2 (CMODC6) Computation and display of the posterior distribution of the regression equation

This module computes the mean and the variance-covariance matrix for the posterior distribution of the regression coefficients and the degrees of freedom for the distribution of the variance of error.

Module 3 (CMODC7) Examination of the posterior distribution of the regression equation

This module allows the user to have the options for examining the posterior distribution of the regression equation. The options are:

1. The distribution of the variance of error
2. The posterior analysis of the regression coefficients
3. Observed and predicted criterion values for the sample data
4. Predictive distribution for chosen predictor values
5. Save the parameters of regression equation in personal file

Module 4 (CMODC8) Display of observed and predicted criterion values for the sample data

This module permits the user to display observed values, predicted values, and residuals of the criterion variable for the sample data.

Module 5 (CMODC9) Save the parameters of the regression equation

This module permits the user to save the parameters of the regression equation on personal file to be used by Component Group 3 for decision analysis.

Model 2 Informative Priors

This model provides a standard Bayesian analysis for simple or multiple regression model assuming normal conjugate priors. The model provides a procedure for the assessment of a Bayes distribution for the regression coefficients. The parameters to be estimated will be divided into three groups:

1. A vector indicating the central tendency of the distribution for regression model.
2. Two numbers, the degrees of freedom and the scale factor for the distribution of the variance of error (residual variance); jointly these two parameters determine the center and the spread of the distribution.
3. A matrix measuring the dispersion of the distribution for the regression coefficients.

This procedure is described in "Interactive Elicitation of Opinion for a Normal Linear Model" (Kadane, Dickey, Winkler, Smith, and Peters, 1978).

Module 1 (CMODC1) Explanation and entry of the dependent and independent variables

This module provides the user with a brief description of the assessment procedure and requests the name of the dependent and independent variables, and the largest and the smallest values of the independent variables.

Module 2 (CMODC2) Estimation of the parameters of the central tendency and the degrees of freedom

This module asks the user to specify the 50th, 75th, and 90th predicted percentiles for various values of the independent variables. The module then estimates the central tendency parameters for the regression coefficients and the degrees of freedom parameter for the variance of error. An opportunity to change the specified percentiles is provided.

Module 3 (CMODC3) Hypothetical data method to estimate the parameters of the scale factor and the dispersion matrix.

This module asks the user to provide the 50th, and 75th predicted percentiles conditioned on some hypothetical data. The module then estimates the scale factor parameter for the variance of error and the dispersion matrix parameter for the regression coefficients.

Module 4 (CMODC4) Display of the parameters of the Bayes distribution for the regression coefficients and the variance of error.

This module displays the parameters of the multivariate t distribution for the regression coefficients and the parameters of the Inverse Chi-squared distribution for the variance of error.

Module 5 (CMODC5) Selection of the independent and dependent variables and entry of prior information.

This module requests the user to select the independent and dependent variables and allows the user to enter prior information from the terminal if it is not stored on the personal file.

Module 6 (CMODC6) Computation and display of the posterior distribution of the regression equation

This module computes the mean and the variance-covariance matrix for the posterior distribution of the regression coefficients and the degrees of freedom for the distribution of the variance of error.

Module 7 (CMODC7) Examination of the posterior distribution of the regression equation

This module allows the user the following options for examining the posterior distribution of the regression equation:

1. The distribution of the variance of error
2. The posterior analysis of the regression coefficients
3. Observed and predicted criterion values for the sample data
4. Predictive distribution for chosen predictor values
5. Save the parameters of regression equation in personal file

Module 8 (CMODC8) Display of observed and predicted criterion values for the sample data

This module permits the user to display observed values, predicted values, and residuals of the criterion variable for the sample data.

Module 9 (CMODC9) Save the parameters of the regression equation

This module permits the user to save the parameters of the regression equation in the personal file to be used by Component Group 3 for decision analysis.

COMPONENT GROUP 3 Decision Theoretic Models

Component 31 (COMP31) Utilities and Expected Utilities

This program controls the activity of Component 31.

Model 1 Assessment of Utilities

This model is concerned with fixed-state methods of assessing a subject's utility function, in which the subject is required to state probabilities that equate certain gambles. Three different assessment procedures are provided in this model to help the user checking the coherence:

1. Fixed-state least-squares assessment
2. Regional-coherence assessment
3. Local-coherence assessment

Module 1 (CMODQ) Explanation of Utility Theory and Entry of Outcomes -- Least-Squares Assessment

This module asks the user to specify the nine outcomes for which he or she wishes to assess the utility function. These do not have to be equally spaced. Also provided is an optional explanation that presents the user with some of the philosophy behind the utility theory.

Module 2 (CMODJ) Entry of Probabilities and Calculation of Initial Utilities -- Least-Squares Assessment

This module asks the user to enter the indifference probabilities for adjacent gambles. The module then asks the user to specify indifference probabilities for additional gambles, forming a set of gambles that will be used to check coherence.

Module 3 (CMODK) Least-Squares Fit of Utilities

This module provides the user with a list of the gambles and indifference probabilities previously specified. An opportunity to change the probabilities is provided. After all changes have been made and the least-squares fit computed, the gambles are again listed, along with the specified probabilities, fitted probabilities, and log-odds differences between the two.

Module 4a (CMODBA) Explanation and Entry of Outcomes -- Regional-Coherence Assessment

This module offers an optional explanation to the user and allows the entry of the outcomes (five, seven, or nine) for the regional-coherence assessment.

Module 5 (CMODB7) Regional-Coherence Assessment

The regional assessment procedure involves the construction of hypothetical choice situations using a for-sure option and a chance option. A for-sure option is, as the name implies, an option that offers the decision maker the certainty of knowing the outcome of a certain decision. A chance option, on the other hand, is one that involves uncertainty about the outcome that will result from choosing it. The chance options used in this assessment procedure involve the possibility of two different outcomes. The three outcomes, one from the for-sure, and two from the chance option, must stand in a particular relationship to each other in order for the choice situation to be meaningful. The two possible outcomes of the chance option must be such that the for-sure option outcome is intermediate in preference to them. An iterative procedure on different situations is executed to test coherence of the user.

Module 6 (CMODBB) Explanation of Local Coherence and Entry of Outcomes

This module allows the user to view an optional explanation of the local-coherence assessment procedure. The user specifies five, seven, or nine outcomes for which a utility function is desired.

Module 7 (CMODB8) Assessment Procedure for Local Coherence

The local-coherence utility assessment procedure can be used with five, seven, or nine different outcomes. In each case the smallest and largest outcomes are arbitrarily assigned utility 0 and 1, respectively. Thus, there are u - 2 (u = 5, 7, or 9) outcomes to which utilities must be assigned. This is then done by using u - 2 different triples of outcomes, selected so that the indifferences probabilities determined for the different triples are sufficient to uniquely assign the u - 2 utilities. There is no provision to check for incoherences by constructing additional hypothetical choice situations. The utilities consistent with the user's preferences are calculated and displayed.

Model 2 Evaluation of Utilities

This model is concerned with the analysis of the utilities. The utilities can be obtained by one of the assessment procedures from model 1 or the user can enter the utilities directly. The model provides the following option for evaluating utilities:

1. Display the assessed utilities and those implied by any fitted parametric function
2. Display the indifference probabilities for choice situations that are implied by the assessed utilities and any fitted parametric functions
3. Display a graph of the assessed utilities or of any utilities determined by a fitted parametric function
4. Fit the normal and Student's t CDF to assessed utilities
5. Fit a generalized beta CDF to assessed utilities
6. Compute the expected utilities

Module 1 (CMODR) Control of the Evaluation of Utilities

This module checks whether the utilities are in the personal file and allows the user to enter the outcomes and utilities.

Module 2 (CMODAR) Examination of Assessed and Fitted Utilities

This module controls the evaluation of the utilities and displays the utilities. Indifference probabilities implied by assessed and fitted parametric utilities for choice situation are computed

Module 3 (CMODL) Fitting Normal and Student's t to Assessed Utilities

This module fits a normal CDF and a Student's t CDF to the user's least-squares (LSQ) utilities. The fitting procedure is actually concerned with fitting the indifference probabilities implied by the LSQ utilities. The log-odds of the indifference probabilities implied by the fitted normal ogive and the LSQ utilities are differenced and the sum of these differences minimized to determine the best fit. After a normal CDF has been fitted to the LSQ utilities, a Student's t fit is attempted using the same procedure.

Module 4 (CMODAO) Fitting a Generalized Beta to Assessed Utilities

This module fits the entire generalized beta CDF to the LSQ utilities. Three different fits are found using the fractile assessment procedure (FASP). Estimates of the 25th, 50th, and 75th percentiles are found using the LSQ utilities and linear interpolation. This module also allows the user to investigate the generalized beta with the following options:

1. Display the assessed and fitted beta utilities
2. Display the indifference probabilities for selected choice situations implied by the assessed and fitted beta utilities
3. Display the indifference probabilities for choice situations implied by the assessed and fitted beta utilities
4. Graph the assessed or fitted beta utilities
5. Select one of the beta fits for the utility function

Module 5 (CMODAT) Expected Utility (Normal, Student's t, and Beta Distributions)

This module allows the user to compute expectation with respect to the following distributions:

1. Normal
2. Student's t
3. Generalized beta

Component 32 (COMP32) Educational and Employment Selection

This program controls the activity of Component 32.

Model 1 Quota-Free Selection (One Group)

In quota-free selection each accept/reject decision is made independently for each applicant. A particular decision is not affected by previous decisions or by those to be made in the future. Each applicant is considered independently of the other applicants.

This model presents a quota-free selection model with a threshold utility function. The threshold utility function implies that each applicant is considered to be in one to two possible states, in this case either qualified or unqualified for the position or program. This threshold utility function can be represented in the following schematic way.

		DECISION	
		Reject	Accept
STATE	Qualified	b	a
	Unqualified	c	d

Ideally, each decision maker would like to have all decisions be correct. And indeed, if the decision maker knew at the time of each decision the status of the applicant, all decisions would be correct. In practice, however, the decision maker can-not know for sure the applicant's status. The best that can be said is that the decision maker knows in a probabilistic sense the applicant's status.

If the decision maker knows the probability p that the applicant is qualified, then according to the principle of maximizing expected utility, the decision should be made by computing the expected utility of both the accept and the reject decisions and taking that action that has the greater expected utility.

Module 1 (CMODBC) Assessment of Threshold Utilities

This module calculates threshold utilities that are appropriate when the outcome of selection can be categorized as either success or failure. The selection decision can lead to one of the following possibilities:

1. An applicant is accepted and succeeds
2. An applicant is rejected who would have succeeded had he or she been selected
3. An applicant is rejected who would have failed had he or she been accepted
4. An applicant is accepted and fails

Module 2 (CMOD90) Determination of Predictor Cut Scores

This module provides selection solutions for both single and multiple predictors with a single criterion. A particular criterion value is specified by the decision maker to mark the minimum criterion indicative of success. The prediction equations are then used to determine the probability that an applicant will be successful. This is taken to be the probability that the applicant is qualified.

For the single predictor case, given the utility function, the prediction equation, and the minimum successful criterion score, the minimum predictor score required for selection is found. The prediction equations may be entered in this module or they could previously have been placed in the personal file by the regression models in Component Group 2.

Model 2 Restricted Selection (Two Groups)

This model presents an assessment procedure for Petersen's model for restricted selection (Novick and Petersen, 1976; Petersen, 1976; and Petersen and Novick, 1976) from two groups with different threshold utility functions. The first step in the analysis is to assess individual threshold utility functions for both groups. This uses the same procedure as in the quota-free selection case. See Component 32, Model 1, Module 1. The utility functions for the two groups are represented below:

	GROUP 1			GROUP 2	
	reject	accept		reject	accept
	B_1	A_1	qualified	B_2	A_2
	C_1	D_1	unqualified	C_2	D_2

Utilities are assessed independently for each group and scaled so that the utility of the most-preferred outcome in each group is 1.00, and the least-preferred outcome in each group is .00.

The next step in the procedure is to assess information from the investigator so that statements may be made involving applicants from both groups at the same time. After assessment, the next step is to determine cut-off scores for both groups such that the selection procedure maximizes the expected utility of the selection process under the constraint that only a specified percentage of the total applicant pool can be accepted. The user may then view the selection of the applicants from the applicant pool. In the selection of the applicants, the analysis will be performed on a maximum of 150 observations from each group.

Module 1 (CMOD91) Assessing Utilities Preparation

This module provides the user with a short explanation and requests the names of the two groups for which a decision must be made.

Module 2 (CMODBI) Assessment of Utilities for Two Groups

This module involves the assessment of utility structures for each group independently. The assessment procedure requires the user to consider an applicant from each group under the assumption that only one applicant can be accepted. The user is told the probability that each applicant will succeed if selected and then indicates which applicant he or she would select. The assessment procedures determine pairs of probabilities for which the user indicates no preference between the two applicants. These so-called indifference probabilities are sufficient to determine a pair of utility structures that reflect the user's preferences.

Module 3 (CMODBE) Scaling the Two Utility Functions

This module scales the two utilities of each group so that the most-preferred outcome in both groups has utility 1 and the least-preferred outcome has utility 0.

Module 4 (CMODBL) Control of the Determination of Cut Scores and the Selection of Applicants

This module determines whether the data set, utilities and prediction equations are in the personal file. It allows the user to enter the utilities from the terminal.

Module 5 (CMODBK) Enter the Regression Equation for Each Group -- Single Predictor

This module provides the options for entering the regression equations for the determination of cut scores. The user can enter the regression equations or sufficient statistics from the terminal if the regression equations have not been stored in the personal file.

Module 6 (CMOD9Y) Determination of Cut Scores for Each Group -- Single Predictor

This module has the user enter the distribution of predictors within each group, and the percentage of the applicant pool that is from each group and the minimum successful score. In the determination of the cut scores, the module displays the cut-off scores and percentages of each group selected for non-user-specified percentages of the total applicant pool to be accepted. The user can then specify percentages and the cut-score and group percentage will be displayed.

Module 7 (CMOD9R) Selection of Applicants

This module reads the predictor information and utilities from the personal file and calculates and displays which applicants would be selected from the applicant pool.

Component 33 (COMP33) Educational and Employment Assignment

This program controls the activity of Component 33.

Model 1 Assignment with Threshold Utilities

This model assumes that each person assigned to a treatment will, upon completion of the treatment, be judged to have succeeded or failed on the basis of the performance on an outcome measure. For each treatment there is a particular value of the criterion variable, called the cut score, that marks the dividing line between success and failure. It is assumed that the same predictors are used to predict the outcome in each of the treatments, even though the outcome measures for the treatments may be different. It is also assumed that the optimal assignment for a person is the treatment that yields the highest expected utility. In this model there may be either two or three treatments. In assigning the applicants to the treatments, the analysis will be performed on a maximum of 200 observations.

Module 1 (CMOD9P) Assessment of Utility Structure

This module begins by asking the decision maker to consider choice situations that involve assigning a person to either treatment 1 or treatment 2. The choice situations present the decision maker with the probabilities that the person will succeed if assigned to treatment 1 and ask the decision maker to specify the probability of success for treatment 2 that would indicate indifference between the two assignments. This procedure can be expanded to three treatments, in which case the procedure is repeated for treatments 1 and 3. Nine sets of probabilities are then listed to be examined. If satisfactory, the utility structure is then presented.

Module 2 (CMODBD) Control of the Determination of Cut Scores and the Treatment Assignment

This module determines if the data set, utilities and prediction equations are in the personal file. It allows the users to enter the utilties from the terminal.

Module 3 (CMODBF) Two Treatment Assignment

This module performs assignment to one of two treatments under the following assumptions. The expected utility of any assignment is assumed to be equal to the sum of the expected utilities of the individual assignments. For any person, prediction equations and threshold utilities are used for each treatment to calculate the expected utility of assignment to each of the treatments. The optimal assignment is assumed to be the one that maximizes the sum of the individual expected utilities. When there are no constraints on the number of persons that can be assigned to each treatment, the optimal assignment is to assign each person to the treatment with the greatest expected utility.

Module 4 (CMOD9Q) Three Treatment Assignment

This module performs assignment to one of three treatments under the following assumptions. In the three treatment case where there are constraints on the number of persons that can be assigned to any one treatment, we have so far only considered the case where a specific number of persons are to be assigned to each treatment. The expected utility of any assignment is assumed to be equal to the sum of the expected utilities of the individual assignments. For any person, the prediction equations and threshold utilities are used to calculate the expected utility of assignment to each of the treatments. The optimal assignment is assumed to be the one that maximizes the sum of the individual expected utilities.

Module 5 (CMODCA) Determination of cut scores for each treatment -- Single Predictor

This module determines the cut scores for each treatment. The user is required to provide the minimum successful criterion score. The regression equation may be entered from the terminal.

Model 2. Assignment with Conditional Utilities

This model is concerned with assigning candidates to one of two "treatments", which may be educational or vocational training programs. The outcome of a treatment is assumed to be shown by the score on a posttest. This score is estimated using a regression equation based on previous experience with the treatment.

The utility of a particular posttest score is conditional; it depends on how the score compares with expectations based on the candidate's score on a pretreatment test. If two persons obtain the same posttest score, there will be more utility in providing the treatment to the person who had the lower pretest score.

The user specifies three pretest scores representing low, middle, and high levels of performance. At each pretest level, four posttest scores are entered to define minimal, satisfactory, good, and superb performance. All posttest scores described by a given label have the same utility. Therefore, it suffices to assess utilities only at the middle level on the pretest. For this level the user provides three additional posttest scores, which lie between those for adjacent labels. Then utilities are assessed at these seven points.

The procedure to define utilities at other pretest scores is illustrated in Figure 2 which is a contour diagram of constant utilities. The posttest scores $x_1,\ldots,x_7$ correspond to minimal,..., superb performance at level M (i.e., middle) on the pretest. Scores y_1, y_3, y_5, y_7 at H (i.e., high) are also specified by the user, corresponding to minimal,..., superb levels of performance. Remaining values are defined by $(y_2-y_1)/(y_3-y_1) = (x_2-x_1)/(x_3-x_1)$, etc.. Then straight lines are drawn to join corresponding points. At any pretest level W, values $z_1,\ldots,z_7$ corresponding to $x_1,\ldots,x_7$ are defined by these lines. The utility of a score z_3 at W is the same as that of x_3 at M. Utilities are assumed to increase linearly between successive scores, e.g., between z_2 and z_3. All scores below z_1 at W have utility 0, and those above z_7 have utility 1. Thus the utility function is defined at all values of pretest and posttest scores.

Each candidate has a pretest score. Therefore, utility is known as a function of posttest score. For each treatment, its regression equation is used to calculate the predictive distribution of posttest score. Then the program calculates expected utility over this distribution. In the absence of limits on enrollment, the candidate should be assigned to the treatment which yields higher expected utility for him/her.

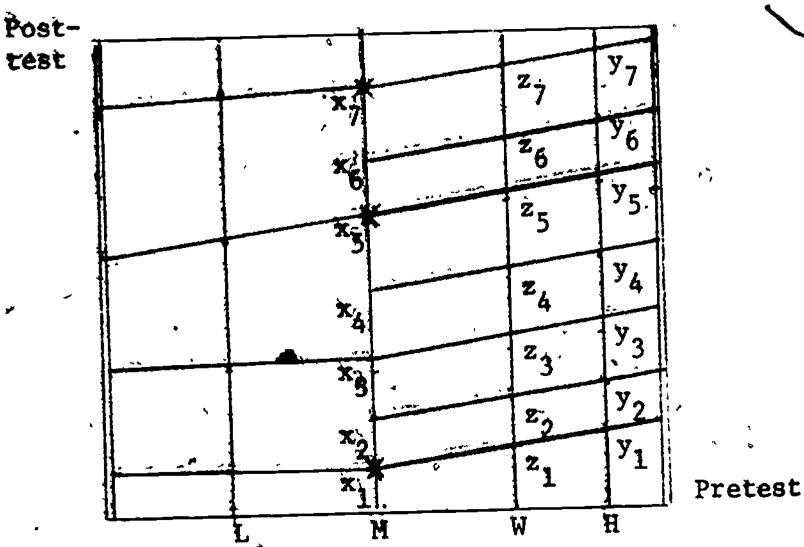


Figure 2

Module 1 (CMODB4) Entry of Pretest and Posttest Scores

This is the first step in assessing conditional utilities. As explained in the overview, the user enters three pretest scores, four posttest scores for each of these, and three additional scores at the middle level on the pretest. At each stage, the user has the option to change any or all the posttest scores entered up to that point.

Module 2 (CMODB9) Choice of Assessment Procedures

This module asks the user to choose between regional and local coherence procedures to assess utilities. The procedures will be explained if desired.

Module 3 (CMODB7) Utility Assessment Using Regional Coherence

Regional coherence is used to assess utilities of posttest scores. For details see Component 31, Model 1, Module 2.

Module 4 (CMODB8) Utility Assessment Using Local Coherence

Local coherence is used to assess utilities of posttest scores. For details see Component 31, Model 1, Module 3.

Module 5 (CMODB6) Collect Information Needed for Assignment

This is the first step in assigning candidates to treatments. The program first checks for availability of data in the personal file. If the data set contains groups, the user is asked to select one for assignment.

Next, the program looks for (at least two) regression equations in the personal file. If these are available, the user is asked which equation should be used with which treatment. If equations are unavailable, the user may exit to the regression component, enter equations from the terminal, or have CADA enter equations based on Data Set 5 in the Catalog.

Finally, scores and utilities are needed. If these are not in the personal file, the user may enter them from the terminal using CMODB5 (Module 6) or assess them using Modules 1 to 4. When all information is available, the program chains to CMODBJ (Module 7).

Module 6 (CMODB5) Entry of Previously Assessed Scores and Utilities

This module is used if utilities have been assessed but are no longer in the personal file. Pretest and posttest scores are entered as in CMODB4 (Module 1), then utilities are typed in.

Module 7 (CMODBJ) Expected Utilities and Assignment to Treatment

Expected utilities of both treatments are calculated for each candidate using the model described in the overview. Averages for the treatments are printed. A listing of all persons is optional. For each person, this shows all scores, both expected utilities, and recommended assignment. The user may specify limits on the numbers of persons that can be assigned to treatments.

COMPONENT GROUP 4 Bayesian Simultaneous Estimation

Component 41 (COMP41) Simultaneous Estimation of Proportions

This program controls the activity of Component 41.

Model 1 Arcsine Transformations

This model provides guidance in estimating simultaneous proportions from exchangeable groups. An arcsine transformation is applied to the proportions as in section 10-1 of Novick and Jackson, 1974. Techniques for the case of equal sample sizes for each proportion are based on "Marginal Distributions for the Estimation of Proportions in m-Groups", (Lewis, Wang, and Novick, 1975). Also included in this model are techniques for analyzing unequal sample sizes. These have been discussed in "The Estimation of Proportions in m-Groups", (Novick, Jackson, and Lewis, 1973).

Module 1 (CMOD8) Prior Distribution

This module helps the user to assess a prior distribution on an exchangeable proportion π. The prior distribution for this model is a beta, and the procedures used to assess it are the same as in Component 21, Model 1, Module 1. Both a fractile assessment procedure and a weighting procedure have been used to help in determining coherence. These are discussed in section 6-5, 6-6, and 6-7 of Novick and Jackson, 1974.

Module 2 (CMODA) Posterior Distributions (Equal Group Sizes)

This module obtains posterior distributions for the probabilities of success (π) in m different exchangeable groups for the case in which sample sizes for all groups are the same. A variance-stabilizing transformation (arcsine) is employed, but all results are given in terms of π values. Joint and marginal point estimates of π for each group are displayed along with selected percentiles.

Module 3 (CMOD9) Posterior Distributions (Unequal Group Sizes)

This module obtains posterior distributions for the probabilities of success (π) in m different exchangeable groups for the case in which sample sizes are not equal. An arcsine transformation is used; however, all results are displayed in the π metric. Posterior joint modal estimates of π for each group, based on the joint mode of the transformed π values, are displayed. Marginal estimates of π for each group also may be obtained.

Component 42. (COMP42) Simultaneous Estimation of Means

This program controls the activity for Component 42.

Model 1 Equal Within-Group Variances

This model assumes that the user has exchangeable beliefs about the group means. The within-group variances are assumed equal. The theory and method are based on Lindley and Smith (1972).

Module 1 (CMOD76) Prior Distribution

This module assists the user in fitting distributions to beliefs about:

1. the grand mean;
2. the mean from a randomly selected group;
3. a randomly selected observation from a group with known mean.

By fitting distrubtions to beliefs about these values, it is possible to infer prior distributions on:

1. between-group standard deviation;
2. within-group standard deviation.

Module 2 (CMOD77) Posterior Distributions

This module obtains posterior distributions for the means of m (maximum 12) different groups under the assumption that the prior beliefs about the means are exchangeable. The user is asked to enter:

1. number of groups;
2. number of observations, mean, and standard deviation for each group.

The sample data are then combined with the prior distributions to form posterior marginal distributions of the group means and linear combinations of group means.

Component 43 (COMP43) Simultaneous Prediction in M-Groups

This program controls the activity for Component 43.

Model 1 Equal Slopes

This model makes available a method for simultaneously estimating regression in m-groups. The slopes are assumed to be identical in the m-groups, but the intercepts may vary. This method is a specialization of the Bayesian linear model developed by Lindley and Smith (1972). The techniques used in this model were developed by Shigemasu (1976) and extended in "A Note on Bayesian Simultaneous Linear Regression with Constant Slopes" (Lewis, 1978). This model assumes exchangeability among the groups studied with a maximum of 10 groups. The analysis will be performed upon a maximum of 25 observations from each group. Up to three predictors and one criterion variable are allowed.

Module 1 (CMOD94) Entry of Data

This module checks the user's personal file for data of the appropriate form. If there are none, the user must use the CADA Data Management Facility (CDMF) to enter or retrieve data. The Sample Regression Data Set, consisting of 10 groups, 4 variables, and 25 observations, may be used. It is described under Component Group 1, Component 12, model 1, module 3.

Module 2 (CMOD62) Least-Squares Setup

This module displays the groups and variables in the user's personal file, asks for criterion variables and predictor variables. It then performs preliminary least-squares calculations.

Module 3 (CMOD9C) Bayesian and Least-Squares Estimates

This module displays Bayesian and least-squares estimates for the intercepts, slopes, and residual standard deviation. There is one intercept for each group and only one set of regression coefficients, since they are assumed equal.

Module 4 (CMOD63) Prediction

This module allows the user to obtain point estimates and probability assessments for each group for future observations given any set of entered values for the predictors.

COMPONENT GROUP 5 Bayesian Full Rank ANOVA

Component 51 (COMP51) Model 1 Factorial Analysis of Variance

This program controls the activity of Component 51. The component provides Bayesian analysis of variance based on a non-informative prior distribution of cell means and residual (within-cell) variance. The Bayesian full rank approach introduced by Novick et. al., in CADA-1978 is adopted and expanded. A design with 1 to 4 factors and 2 to 32 cells can be analyzed. Main effect and interaction parameters are computed as linear combinations of cell means. The posterior multivariate t distribution of these parameters may be studied. In particular posterior means and standard deviations of these parameters are available as well as credibilities. (The credibility of a hypothetical parameter point is the probability content of the smallest highest density region (HDR) containing that point. This method of analysis is based on ideas presented in Bayesian Inference in Statistical Analysis (Box & Tiao, 1973).)

Model 1 Place a 'Breakdown' file of summary statistics on the personal file.

Module 1 (CMODD1) Assembly of Summary Statistics

This module controls the activity of the model. The user has these options:

1. Display summary data already on personal file.
2. Compute, display and file summary data from raw data on personal file. Note: Raw data is destroyed.
3. Type in summary data at the terminal, display and file it. Note: Any data on file is destroyed.

Module 2 (CMODD2) Entry of Summary Statistics from the Terminal

This module is controlled by CMODD1. It provides option 3 of Model 1.

Model 2 Compute Main effects and Interactions

This model allows the user to select linear combinations of cell means (main effects and interactions). For each factor the user may select the default definition of main effects (successive differences) or may type in contrast coefficients. Interactions are automatically generated as 'products' of main effects. It is also possible to select 'simple' main effects although this option is offered only when the user is returning to Model 2 from Model 3. Simple main effects are explained in Chapter 5 of Statistical Principles of Experimental Design (B. J. Winer, 1971).

Module 1 (CMODD3) Generation of Main Effects and Interactions

This module asks the user for 'main effect' linear combinations of levels of each factor (default or user-supplied), generates from these the 'interaction' linear combinations, computes the parameters of the posterior distribution of main effects and interactions, and files these parameters.

Module 2 (CMODD4) Tutorial on Main Effects and Interactions

This module is controlled by CMODD3. It provides tutorials on main effects and interactions.

Module 4 (CMODDM) Storage of Parameters

This is an 'invisible' utility module used to append the posterior parameters to the summary data file.

Model 3 Analyze Posterior Distributions

This model allows the user to study the posterior multivariate t distribution of the parameters. The user is guided through a series of steps.

Step 1: Posterior means and S.D.'s of parameter are displayed.
Step 2: User is allowed to condition on or marginalize any parameters.
Step 3: User is offered these options:

a) Redisplay the (now conditionalized) posterior means and S.D.'s.
b) Compute the probability content of the smallest HDR containing user-specified values of the active parameters (neither conditioned nor marginalized).
c) End this 'round' of analysis.
   i) Add more conditions or marginalizations.
   ii) Return to Model 2 to redefine main effects and interactions.
   iii) End Model 3.

Module 1 (CMODD5) Analysis of Posterior t-distributions of Effects

This module controls the activity of Model 3.

Module 2 (CMODD6) Calculation of Probability Content of Smallest HDR

This module is controlled by CMODD5. It computes and displays the probability content of a highest density region (HDR).

Module 3 (CMODD7) Tutorial on Highest Density Regions

This module is controlled by CMODD5. It provides a tutorial on HDR's.

Component 52 (COMP52) Analysis of Repeated-Measures Designs

This program controls the activity of Component 52. The component is a duplicate of Component 61. It provides analysis of repeated-measures designs and analysis of covariance. Informative priors may be used. It can also perform the analyses provided by Component 51 for model 1 designs with 27 cells or fewer. (But note that Component 52 cannot process a summary data set generated by Component 51 -- the user will have to reload the raw data.) The capacity of this component is:

$$M + B \leq 28$$

$$(M+1) \times (B+1) \leq 120$$

where
M = # of measures per subject
and
B = # of cells "between" subjects.

There can be up to 4 "within" subject factors and up to 7 factors in all "between-" and "within-" subjects.

COMPONENT GROUP 6 Bayesian Multivariate Analysis

Component 61 (COMP61) Bayesian Full-Rank Multivariate Analysis of Variance

This program controls the activity of Component 61.

This component provides Bayesian multivariate analysis of variance of designs involving up to seven factors of which no more than four can be "within subjects"..

The user may select a non-informative prior (Box and Tiao, 1973, p, 426) or may type in the parameters of an informative, conjugate matric T prior. The user may transform the cell means into main effects and interactions, as in Component 51, or keep the cell mean parametrization. The user may then examine the posterior, matric T distribution of the parameters. This method of analysis is based on ideas presented in Chapter 8 of Box and Tiao, 1973. For more details run Model 7 (Tutorial) of this component.

Summary data computed by this component are available to the data management component for transfer to a permanent file (if the user has a password). These data can be reloaded for reanalysis in a later session. After reloading the analysis can begin where it left off.

This component can perform the following analyses: Factorial ANOVA and MANOVA, Repeated-Measures ANOVA and MANOVA, Growth Models, Analysis of Covariance and some Multivariate Analysis of Covariance. Its limitations are:

1. No more than 5 dependent variables plus within subjects factors combined
2. No more than 8 factors plus dependent variables combined
3. No more than 28 cells "between subjects" plus measurements per subject combined
4. No more than 120 cells in all

Model 1 Put Description of Layout of Experiment on File

This model determines the layout of the data; i.e., the name, number of levels and type (between or within subjects) of each factor and the names of the dependent variables. This information is placed on the personal file in a group named "FACTOR". The functions of this model are carried out by the following modules:

Module 1 (CMODDO) Determine Layout of Design to be Analyzed

This module controls the activity of Model 1. The options it offers depend upon what type of data it finds on the personal file:

1. With raw data the user is shown the variable names and group names (if any) and asked to specify the role of each variable in the analysis: between-subjects factor, within-subjects factor, dependent variable or not in the analysis. This module then reads the data and determines the number of levels of each factor.
2. With summary data CADA reads layout information from the personal file, shows it to the user and asks if these data are to be analyzed.
3. If there are no data on file, or the filed data are not to be analyzed, the user is invited to type in all the layout information.

Module 2 (CMODDF) Continuation of CMODDO

This module collects layout information from the user (option 2).

Model 2 Put Prior Information on File

This model allows the user to enter the parameters of a prior distribution and places them on file. The user has these options:

1. Non-informative prior
2. Informative - conjugate prior

The activities of this model are carried out by these modules.

Module 1 (CMODDA) Collect Z'Z Matrix from the Terminal

This is a utility module which is used here to collect parameters of the prior distribution from the user under option 2.

Module 2 (CMODDM) Add Group to Personal File

This is a utility module used here to place the prior parameters on the personal file in a group named "PRIOR".

Model 3 Put Summary Statistics on File

This model collects or computes summary statistics and puts them on the personal file. There are two options:

1. Compute summary statistics from raw data. (Note: These raw data must have been on the personal file before this component was entered).
2. Type in summary statistics at the terminal. The summary statistics are: cell n's and means and pooled, within-cell standard deviations and correlations.

After execution of Model 3, the design information and summary statistics are on the user's personal data file and may be transferred to a permanent file any time before exiting from CADA. The following modules perform the activities of this model:

Module 1 (CMODDA) Collect Z'Z Matrix from the Terminal

This is a utility module which is used here to collect summary statistics from the terminal under option 2.

Module 2 (CMODDE) Compute Summary Statistics from Raw Data on File

Module 3 (CMODDM) Add Group to Personal File

This is a utility module used here to place summary statistics on the personal file in a group named "Z'Z".

Model 4 Compute Posterior Distribution of Cell Means

This model computes the parameters of the posterior matric T distribution of the cell means.

Module 1 (CMODDB) Mix Prior and Posterior Distributions

This module computes the parameters of the posterior matric T distribution of the cell means using the parameters of the prior distribution and the summary statistics on file. (Note: For the non-informative prior no computation takes place since the parameters of the posterior matric T are the summary statistics).

Module 2 (CMODDM) Add Group to Personal File

This is a utility module used here to place the posterior parameters on the personal file in a group named "POST".

Model 5 Transform Cell Means to Effects and Interactions

This model transforms the posterior distribution of the cell means to that of the main effects and interactions. The user is shown the name of each factor and asked to say how its main effects are to be defined. The options for each factor are:

1. Standard effects - i.e., effect #1 = level 2 - level 1, effect #2 = level 3 - level 2, etc.
2. Normed orthogonal polynomials
3. User-supplied contrast coefficients or
4. Don't define effects - break down the analysis according to levels of this factor

This model constructs the interaction contrast coefficients as products of main effect coefficients, and transforms the parameters of the posterior distribution as in formula (8.4.44) of Box and Tiao, 1973. For more details see Woodworth, 1979 and the tutorials in Model 7. The activities of this model are carried out by these modules.

Module 1 (CMODDG) Select Definitions of Effects. Build Transformation Matrix

This module collects effect options from the user and constructs "between" and "within" transformation matrices. (P and Q of Box and Tiao, 1973 formula 8.4.44).

Module 2 (CMODDG) Continuation of CMODDC: Calculate Transformed Parameter Matrices

This module calculates the parameters of the matric T distribution of the effects and interactions.

Module 3 (CMODDM) Add Group to Personal File

This is a utility module used here to place the parameters of the transformed distribution on the personal file in a group called "T'name T" where name refers to the last set of parameters on the data file, i.e., name = PRIOR if an informative prior has been entered but the summary data have not, name = Z'Z if a non-informative prior was used and name = POST if an informative prior has been combined with the summary data.

Model 6 Examine Matric T Distribution of Parameters

This model allows the user to examine a matrix T distribution (prior or posterior) of cell means or effects and interactions. The distribution may be conditionalized or marginalized. The distribution analyzed is the last one the user has placed on the data file. The following modules carry out the function of this model:

Module 1 (CMODDD) Examine Matric T Distribution

This module controls the activity of this model, offering options:

1. Display means and standard deviations of MANOVA parameters. (Note: The standard deviation here is the standard deviation of the matric T distribution of the MANOVA parameters (cell means or effects and interactions) - it is analogous to

the STANDARD ERROR of an estimate in classical statistics.)

2. Alter the status of MANOVA parameters
   i) remove all conditions/marginalizations (i.e., reload parameters from file)
   ii) add conditions or marginalizations
3. Compute the credibility of a hypothetical set of parameter values typed in by the user. That is, compute the probability content of the smallest highest density region (HDR) containing the hypothetical parameter values. (Note: With a non-informative prior this probability is generally equal to 1 minus the significance level attained by Wilk's lambda test of the "hypothesis" that the MANOVA parameters equal the hypothetical values typed in by the user, although there are minor differences in the degrees of freedom for error for classical and Bayesian analyses under some patterns of conditioning.)

Module 2 (CMODDH) Conditionalize and/or Marginalize

This module displays the names of the MANOVA parameters and asks the user whether to keep, conditionalize or marginalize each parameter -- certain mathematical restrictions on the pattern of conditioning and marginalization are enforced. See the tutorials in Model 7 or Woodworth, 1979, for more details.

Module 3 (CMODDI) Sweep Operator

This is a dual-purpose module which either collects the user's conditional parameter values and computes parameters of the conditional matric T distribution (option 2.ii) of the remaining parameters or collects the user's hypothetical parameter values, θ_{HYP}, (option 3) and computes $U(\theta_{HYP})$ (Box and Tiao, 1973 Formula 8.4.49 or Woodworth, 1979, pp. 390-391).

Module 4 (CMODDJ) Calculate Content of HDR

This module computes the probability content of the smallest HDR containing θ_{HYP}. It uses Rao's

F-approximation to the distribution of $U(\theta_{HYP})$ (Woodworth, 1979, page 391).

Model 7 Tutorial

This model provides tutorials on analysis of multivariate experimental designs. It offers instruction on:

1. Factors and variables
2. Raw data formats
3. Summary statistics
4. Main effects and interactions
5. Breakdowns on a factor
6. Matric T distribution — conditioning and marginalizing
7. HDR's

Module 1 (CMODDK) MANOVA Tutorials 1, 2 and 3

Module 2 (CMODDL) MANOVA Tutorials 4 through 7

COMPONENT GROUP 7 Elementary Classical Statistics

Component 71 (COMP71) Frequency Distributions

This program controls the activity for Component 71.

Model 1 Absolute-Frequency Histograms

Module 1 (CMOD43) Absolute-Frequency Histograms

This module allows the user to obtain a histogram of any variable in the personal file. The user specifies the number of intervals (up to 15) and whether the user or the system is to select the interval boundaries.

Model 2 Contingency and Expectancy Tables

Module 1 (CMOD14) Two-way Contingency and Expectancy Tables

This module allows the user to view overall contingency and expectancy tables, or independent expectancy tables on any two variables in the personal file.

Component 72 (COMP72) Summary Statistics

This program controls the activity for Component 72.

Model 1 Summary Statistics

Module 1 (CMOD13) Summary Statistics

This module allows the user to obtain a variety of descriptive statistics (means, standard deviations, percentiles, etc.), a variance-covariance matrix, and a correlation matrix.

Component 73 (COMP73) Graphic Displays

This program controls the activity for Component 73.

Model 1 Absolute-Frequency Histograms

Module 1 (CMOD43) Absolute-Frequency Histograms

This module allows the user to obtain a histogram of any variable in the personal file, specifying the number of intervals (up to 15) and whether the user or the system is to determine the interval boundaries.

Model 2 Bivariate Plots

Module 1 (CMOD42) Bivariate plots

This module allows the user to obtain a bivariate scatter plot of any two variables in the personal file.

Component 74 (COMP74) Regression

This program controls the activity for Component 74.

Model 1 Simple or Multiple Linear Regression

Module 1 (CMODB1) Entry of Data

The module asks the user to provide the dependent and independent variables and computes the regression coefficients, standard errors, F, t, R^2 and residuals.

Module 2 (CMODB2) Regression Analysis

The module controls the regression analysis. The user can specify residuals and predicted values via tables, normal probability plots of residuals, or bivariate scatter plots.

Module 3 (CMOD42) Bivariate Plots

This module allows the user to obtain a bivariate scatter plot for the variables in the regression equation or the residuals (RESIDL) or predicted values (Y-HAT).

Module 4 (CMOD9I) Normal Probability Plot

This module allows the user to view a normal probability plot of the residual from a regression. This module is also used in the EDA component group.

Module 5 (CMOD15) Transformations

This module allows the user to transform the values in the personal file, after a regression analysis, and before another regression analysis. This module calls CMODE0, CMODE1, and CMODE2, which are described in Component Group 1.

COMPONENT GROUP 8 Exploratory Data Analysis (EDA)

Component 81 (COMP81) Univariate EDA

This program controls the activity of Component 81.

Model 1 Regular CRT Applications

This model presents univariate techniques that may be displayed on standard 24-line by 80-character CRT terminals. There are two help modules that explain univariate EDA and describe techniques which can be used to answer questions that are often addressed in data analysis. Each technique also has an optional explanation that provides instructions, examples, and uses for the technique. This model has adapted techniques from Exploratory Data Analysis (Tukey, 1977), Data Analysis and Regression (Mosteller and Tukey, 1977), and Methods for Statistical Data Analysis of Multivariate Observations (Gnanadesikan, 1977). An in-depth explanation of this component appears in Conversational Exploratory Data Analysis (Isaacs, 1978). The user has the following options:

Overview

1. Descriptions of methods in Exploratory Data Analysis (EDA)
2. Questions involved in EDA and associated techniques

Techniques

3. Box Plot
4. Stem-and-Leaf Diagram
5. Empirical Probability Density Function (EPDF)
6. Smoothed EPDF
7. Empirical Cumulative Distribution Function
8. Normal Probability Plot

Alteration

9. Standardization
10. Transformation (Reexpression)
11. Trimming of Extremes
12. Selection of a new variable
13. Retrieval of original data (after alteration)

Module 1 (CMODA3) Explanation of Univariate EDA

This module gives the user a short general explanation of univariate EDA. The user is also introduced to those aspects of data which are usually addressed by EDA. Upon completion of this module, control is passed to Module 2, which describes in detail the information available through EDA.

Module 2 (CMODA4) Questions in Univariate EDA

This module lists questions that need to be answered to explore univariate data. Each question may be addressed in detail, if the user wishes. The question is explained and the techniques that best answer it are shown. A summary table that ranks the techniques best suited to answer each question concerning the aspects of the data is also provided. The following questions are addressed:

1. What are the extreme values of the data?
2. What is the central tendency?
3. How variable are the observations?
4. How many modes do the data have?
5. Are the data symmetric or skewed?
6. Are the data flat or peaked?
7. Are there outliers?
8. Are there gaps?
9. Are the data normally distributed?
10. Can abnormal appearances be reduced by transformation?
11. What happens after standardization?
12. What is the result when the outliers are trimmed?
13. Is there a ceiling or floor effect?

Module 3 (CMOD9D) Selection of Variable

This module allows the user to choose for examination one of the variables in the personal file. If the summary statistics option is requested in this module, it leads the user to Module 15.

Module 4 (CMOD9E) Option List

This module provides a list of available options. The list is divided into three sections:

overview, techniques, and alteration. The options are listed in the overview of this model.

Module 5 (CMOD99) Box Plots

This module first presents the user with an optional explanation of box plots as they have been implemented. This explanation presents sample box plots that are explained in depth regarding potential uses. Those characteristics of univariate data that may be seen from box plots are also listed. The box plot of the variable specified by the user gives a visual impression of the extremes, range, quartiles, mean, median, and standard deviation.

Module 6 (CMOD98) Stem-and-Leaf

This module first presents the user with an optional explanation of stem-and-leafs as they have been implemented. Included in the explanation are sample stem-and-leaf plots, their uses, and the aspects of the data they display. A stem may appear on one, two, or five lines, depending on the data. All leaves are single digit, which allows the user to get a picture of the data. This provides the user with two significant digits of accuracy and a visual display of the observations.

Module 7 (CMOD9H) Empirical Probability Density Function

This module first presents an optional explanation of the empirical probability density function (EPDF), with examples, uses, and particulars of this technique as implemented. This technique divides the vertical axis into 20 intervals and plots the number of observations in each interval, as in a histogram.

Module 8 (CMOD9J) Smoothed Empirical Probability Density Function

This module presents an optional explanation of a smoothed EPDF as implemented. This technique allows the user to smooth the EPDF any number of times to obtain a curve that does not show idiosyncrasies. The method of running means is used for the smoothing.

Module 9 CMOD9A) Empirical Cumulative Distribution Function

This module presents an optional explanation of the empirical cumulative distribution function (ECDF). Examples, uses, and techniques are displayed and described, The horizontal axis and vertical axis are both divided into 60 intervals. An ECDF is a plot of the I-th ordered observation on the vertical axis versus (I - .5)/N on the horizontal axis. A maximum of 60 observations may be plotted, so an algorithm is used to select observations. Three different symbols (. * ') are used to plot observations. In this way the vertical axis is divided into 60 intervals while using only 20 lines. The . appears lower than the *, which appears lower than the '.

Module 10 (CMOD9I) Normal Probability Plot

This module presents an optional explanation containing uses, examples, and techniques of this implementation of normal probability plots. In a normal probability plot, quantiles of the univariate data on the vertical axis are plotted against quantiles of a standard normal distribution on the horizontal axis. The vertical axis is divided into 20 intervals, ranging from the lowest to the highest value. The horizontal axis is divided into 60 intervals from -3.0 to 3.0. If the plot is an approximately straight line, then the data probably came from a normal distribution.

Module 11 (CMOD9K) Standardization

This module allows the user to standardize the univariate data into values with a mean of zero and a standard deviation of one. These data can then be examined by any of the univariate EDA techniques. This module provides only temporary alteration of the data. If the user wishes a permanent change of the personal file, this module is not appropriate. For permanent changes use Component 13, Model 1.

Module 12 (CMOD90) Transformation (reexpression)

This module allows the user to transform the variable that is being examined. The following transformations may be used:

1. Power or linear
2. Logarithm to user-specified base
3. Log-odds, with base of logarithm given by user
4. Arcsine
5. Log-odds (base 10) transformation of rank

Explanation and appropriate uses of each transformation may be obtained. Once a transformation has taken place, the data may not be transformed again without retrieval of the original variable or selection of another variable. This module provides only temporary alteration of the data. If the user wishes a permanent change of the personal file, this module is not appropriate. For permanent changes use Component 13, Model 1.

Module 13 (CMOD9W) Trimming of Extremes

This module allows the user to trim extreme values from either or both ends of the univariate data. Up to 15 percent may be trimmed off from each end. This module provides only temporary alteration of the data. If the user wishes a permanent change of the personal file, this module is not appropriate. For permanent changes use Component 12, Model 2.

Module 14 (CMOD9N) Retrieval of Original Data (after Alteration)

This module allows the user to obtain unaltered data that were being examined before standardization, transformation, or trimming took place.

Module 15 (CMOD13) Summary Statistics

This module provides the user summary statistics for all variables in the personal file. This module works independently of the modules for temporary alterations. That is, this module always reports the summary statistics of the original data. See the explanation of Component 72.

Component 82 (COMP82) Bivariate EDA

This program controls the activity of Component 82.

Model 1 Regular CRT Applications

As in univariate EDA, the major goal of bivariate exploratory data analysis is to give the investigator an intuitive impression of data. In bivariate EDA, the investigator is interested in the aspects of two variables. Prevalent trends, tendencies, and an overall picture of the data provide a description of the two variables for the investigator. Bivariate EDA also gives the investigator an insight into particulars of the data. These particulars consist of outliers, gaps, secondary modes, and other irregularities. This mixture of general and particular details provides the investigator with a description of the data being examined. Bivariate data are examined from three different points of view: independently, jointly, or conditionally. Thus, the user can use these to form a more complete description of the data. Bivariate EDA is described in greater detail in Conversational Exploratory Data Analysis (Isaacs, 1978). The user has the following options:

Overview

1. Overview of bivariate exploratory data analysis (EDA)
2. Questions in bivariate EDA and associated techniques

Techniques

3. Summary statistics
4. Scatter plot
5. Schematic plot
6. Univariate EDA conditional on X
7. Conditional expectation (regression) plot

Alteration

8. Standardization
9. Transformation (reexpression)
10. Trimming
11. Selection of new variables
12. Retrieval of original data (after alteration)

Module 1 (CMODA5) Overview of Bivariate Exploratory Data Analysis

The bivariate overview module presents the investigator with several frames of text that provide direction and guidance in the use of bivariate EDA. The first two frames present an explanation of the major goal of bivariate EDA: to help the investigator achieve an intuitive impression of data. These frames also present some of the philosophical underpinnings of the development of bivariate EDA. A list of the aspects of data to be considered is presented. These aspects are: extremes and magnitude, central tendency, variability, modality, skewness, kurtosis, normality, outliers or gaps, and relationship between the two variables. These aspects can be considered independently, jointly, and conditionally to give the investigator a full description of the data. Module 1 automatically leads into Module 2.

Module 2 (CMODA6) Questions Involved in Bivariate EDA and Associated Techniques

This module provides the investigator with a list of questions that are most likely to be addressed in the exploration of bivariate EDA. These are provided in the order in which the questions would typically be addressed. A list of these questions follows:

1. What are the extreme values of the data?
2. What is the central tendency?
3. How variable are the observations?
4. How many modes do the data have?
5. Are the data symmetric or skewed?
6. Are the data flat or peaked?
7. Are there outliers?
8. Is there a ceiling or floor effect?
9. Are the data normally distributed?
10. How strong is the relationship between the two variables?
11. Is the relationship a smooth trend or are there distinct, separate clusters?
12. Is the relationship linear?
13. Does the magnitude of the variability of one variable change across the range of the other (heteroscedasticity)?

A summary of the questions and techniques that best address them is then presented. This summary indicates whether a technique addresses the data independently, jointly, or conditionally. This summary table provides a large amount of information in one frame. An investigator may plan necessary analyses from this summary.

Module 3 (CMOD9V) Selection of Variables

This module allows the user to select the two variables of current interest from those in the personal file. If summary statistics option is requested, the program proceeds to Module 24.

Module 4 (CMODA1) Option List

This module presents the user with the option list from which the user can select the desired technique. The options available are listed in the overview of this model.

Module 5 (CMODA9) Summary Statistics

The summary statistics module of bivariate EDA provides an investigator with statistics for both variables. They help the user to gain insights into each variable independently. The following statistics are provided: high, low, mean, 25th percentile (Q1), 50th percentile (Q2 or median), 75th percentile (Q3), standard deviation, variance, and interquartile range (Q3 - Q1). These statistics help the investigator to examine the following aspects of each variable by itself: extremes, central tendency, variability, and skewness. This option also provides the user with a short explanation of the statistics and how they should be interpreted: for example, skewness can be judged by comparing the difference from the 25th percentile to the median against the difference from the median to the 75 percentile.

Variance/Covariance Matrix and Correlation Matrix are also provided.

Module 6 (CMODA2) Explanation of Scatter Plot

This module provides an explanation of a scatter plot, example, and uses. It then returns the user to Module 7, where the actual scatter plot of the data is produced.

Module 7 (CMODA7) Scatter Plot

A scatter plot of bivariate data provides the investigator with a joint plot of the two variables. The ranges of the variables on both X and Y axes are divided into 20 intervals to form 400 cells. The number of observations that fall into each cell is shown on the grid. Special symbols are provided if 10 or more observations fall in a cell. The following shows the symbol plotted for a given number of observations in a cell:

Number of Observations	Symbol Plotted
0	blank
1 through 9	digit
10 through 19	&
20 through 29	*
30 or greater	$

A scatter plot shows the investigator a two-dimensional picture of data. A third dimension can be seen by the number of observations in a cell. The higher the number of observations, the greater the density at those values of the X and Y variables. Flat areas and peaks can be seen from the values of the third dimension.

Module 8 (CMODA8) Explanation of Schematic Plot

This module provides an explanation of a schematic plot with examples and uses. It then leads the user to Module 9, where the actual schematic plot of the data is produced.

Module 9 (CMODAI) Schematic Plot

This technique provides the investigator with a sensitive tool for studying the trend of the Y-variable as the X-variable increases. The display consists, in effect, of side-by-side conditional box plots. The X-variable is divided into four inter-quartile intervals or 'slices' containing approximately equal numbers of observations. For each X-slice, the box plot of the Y- values in the slice is shown. Besides four conditional box plots of the Y-variable, the box plot of the X-variable is provided also. The box plot of the X-variable is rescaled so that it has the same range as the Y-variable. The scale for the Y-box plots is provided at the left margin. The locations of the Y-box plots are not meaningful. They are equally spaced. Conditional summary statistics are also provided.

Module 10 (CMODAK) Definition of Conditional

This module allows the user to define a conditional on the X-variable. The data on the Y variable are then saved in the personal file. Control is then passed to Module 11.

Module 11 (CMODAS) Conditional Selection

This module allows the user to select any of the univariate techniques to examine the conditional data. A list of the univariate techniques follows:

1. Box plot
2. Stem-and-leaf
3. Empirical probability density function (EPDF)
4. Smoothed EPDF
5. Empirical cumulative distribution function (ECDF)
6. Normal probability plot

Module 12 (CMOD99) Box Plot

This module allows the user to view a box plot of the conditional of Y on the range of X defined in Module 10. For an explanation see Component 81, Model 1, Module 5.

Module 13 (CMOD98) Stem-and-Leaf

This module allows the user to view a stem-and-leaf of the conditional of Y on the range of X defined in Module 10. For an explanation see Component 81, Model 1, Module 6.

Module 14 (CMOD9H) Empirical Probability Density Function

This module allows the user to view an EPDF of the conditional of Y on the range of X defined in Module 10. For an explanation see Component 81, Model 1, Module 7.

Module 15 (CMOD9J) Smoothed EPDF

This module allows the user to view a smoothed EPDF of the conditional of Y on the range of X defined in Module 10. For an explanation see Component 81, Model 1, Module 8.

Module 16 (CMOD9A) Empirical Cumulative Distribution Function (ECDF)

This module enables the user to view an ECDF of the conditional of Y on the range of X defined in Module 10. For an explanation see Component 81, Model 1, Module 9.

Module 17 (CMOD9I) Normal Probability Plot

This module permits the user to view a normal probability plot of the conditional of Y on the range of X defined in Module 10. For an explanation see Component 81, Model 1, Module 10.

Module 18 (CMODAP) Retrieval of Intermediate Data after Conditionalization

This is an intermediate module that is invisible to the user. Its function is to reset the personal file for bivariate analysis after a conditional has been analyzed.

Module 19 (CMODB3) Conditional Expectation Plot

A conditional expectation plot provides a graph of the conditional means of the Y-variable given a set of intervals of the variable on the X-axis. The investigator is asked to enter the number of intervals into which the X-axis is to be divided. The minimum number of intervals is 10, and the maximum is 60. Then, the Y-values for any observations whose X-values fall in an interval are averaged and a symbol plotted opposite the appropriate Y-value and above the appropriate X-interval.

This symbol will also give an indication of the number of observations upon which the conditional mean was based. The following symbols were used:

Number of Observations	Symbol Plotted
1 through 9	digit
10 through 19	&
20 through 29	*
30 or greater	$

Module 20 (CMODAJ) Standardization

In bivariate standardization, the user standardizes both variables to a mean of 0 and standard deviation of 1, thus permitting analysis of the data without scale or location effects. Once standardization has taken place, new variables must be selected or the original data retrieved before the standardization option may be used again.

This module provides only temporary alteration of the data. If the user wishes a permanent change of the personal file, this module is not appropriate. For permanent changes, use Component 13, Model 1.

Module 21 (CMODAL) Transformation

Bivariate transformation allows the investigator to transform one or both of the variables being examined. The same transformations are available in bivariate EDA as under univariate EDA. The transformations are:

1. Power or Linear
2. Logarithm to specified base
3. Log-odds to specified base
4. Arcsine
5. Log-odds of rank, base 10

Once a bivariate transformation has taken place, the data may not be transformed again without selection of new variables or retrieval of the original data. The various transformation options allow an investigator to transform a wide variety of data. Different transformations may be applied to the X and Y variables.

This module provides only temporary alteration of the data. If the user wishes a permanent change of the personal file, this module is not appropriate. For permanent changes, use Component 13, Model 1.

Module 22 (CMODAM) Trimming

In bivariate trimming, the investigator is allowed to trim either or both of the variables. The investigator may first trim up to 15 percent of the observations from the high and/or low end of the X variable. The percentages trimmed off the two ends may be different. After this trimming, the new number of observations is presented and the Y variable may then be trimmed in a like manner. (In this case, the percentages to be specified refer to the observations that remain after trimming on X. This allows the investigator a wide latitude in what is being trimmed.)

This module provides only temporary alteration of the data. If the user wishes a permanent change of the personal file, this module is not appropriate. For permanent changes, use Component 12, Model 2.

Module 23 (CMODAN) Retrieval after Alteration

The retrieval option of bivariate EDA allows the investigator to get back the data that were available before being altered by standardization, trimming, and/or transformation. The names of the original variables and the number of observations are presented to the user before being returned to the menu.

Module 24 (CMOD13) Summary Statistics

This module provides summary statistics of all the variables in the personal file. See the explanation of Component 81, Model 1, Module 15.

disribution. In addition, percentiles can be obtained for truncated t distributions. Subroutine TDTR is used for calculation of the CDF.

Module 3 (CMODD) Inverse Chi Distribution

This module enables the user to study an inverse chi distribution after entering the degrees of freedom and the scale parameter. Options available are the same as for the normal distribution. Probabilities and percentiles are calculated by converting the inverse chi into a chi-square. The module uses subroutines CSQDTR for calculation of chi-square CDF and ICHDR for calculation of HDR.

Module 4 (CMODN) Inverse Chi-Square Distribution

This module provides information about an inverse chi-square distribution after the degrees of freedom and the scale parameter are entered. Options are the same as for the normal distribution. Reciprocal of the inverse chi-square is used for computation of probabilities and percentiles. Subroutines employed are CSQDTR for chi-square CDF and ICSQHR for calculation of HDR.

Module 5 (CMODO) Chi-Square Distribution

This module is used to study a chi-square distribution after entering the degrees of freedom and the scale parameter. The same options are available as with the normal distribution. The module uses subroutine CSQDTR for calculation of probabilities and CSQHDR for highest density regions.

Module 6 (CMODB) Beta Distribution

This module allows the user to investigate a two-parameter beta distribution after the parameters have been entered. The same five options as in the normal distribution are available. Subroutines used are BDTR for calculation of CDF and BHDR for highest density regions.

Module 7 (CMODF) Behrens-Fisher Distribution

This module enables the user to study a Behrens-Fisher distribution, which is the distribution of the difference of two independent t distributions. Two disribution. In addition, percentiles can be obtained for truncated t distributions. Subroutine TDTR is used for calculation of the CDF.

Module 3 (CMODD) Inverse Chi Distribution

This module enables the user to study an inverse chi distribution after entering the degrees of freedom and the scale parameter. Options available are the same as for the normal distribution. Probabilities and percentiles are calculated by converting the inverse chi into a chi-square. The module uses subroutines CSQDTR for calculation of chi-square CDF and ICHDR for calculation of HDR.

Module 4 (CMODN) Inverse Chi-Square Distribution

This module provides information about an inverse chi-square distribution after the degrees of freedom and the scale parameter are entered. Options are the same as for the normal distribution. Reciprocal of the inverse chi-square is used for computation of probabilities and percentiles. Subroutines employed are CSQDTR for chi-square CDF and ICSQHR for calculation of HDR.

Module 5 (CMODO) Chi-Square Distribution

This module is used to study a chi-square distribution after entering the degrees of freedom and the scale parameter. The same options are available as with the normal distribution. The module uses subroutine CSQDTR for calculation of probabilities and CSQHDR for highest density regions.

Module 6 (CMODB) Beta Distribution

This module allows the user to investigate a two-parameter beta distribution after the parameters have been entered. The same five options as in the normal distribution are available. Subroutines used are BDTR for calculation of CDF and BHDR for highest density regions.

Module 7 (CMODF) Behrens-Fisher Distribution

This module enables the user to study a Behrens-Fisher distribution, which is the distribution of the difference of two independent t distributions. Two options are available for entering parameters. The user may enter degrees of freedom, mean, and scale parameter for each t distribution. In the other option one specifies the two degrees of freedom, angle ψ, mean and scale for the Behrens-Fisher. Options available are the same as for the normal distribution. Calculations are carried out by approximating the distribution with a Student's t (Patil, 1964).

Module 8 (CMODZ) F Distribution

This module allows the user to study an F distribution after the two degrees of freedom (ν_1, ν_2) and two scale parameters (λ_1,λ_2) are entered. (ν_1,λ_1) refer to the inverse chi-square in the denominator. The same options as for the normal distribution are available. Probabilities and percentiles are calculated by transforming F into a beta variable.

Module 9 (CMODV) Binomial Distributions

This module enables the user to study a binomial distribution with specified size and process parameters. Two options are available:

1. Probabilities that the number of successes will be less than X, equal to X, and greater than X.

2. Probability that the number of successes will be at least X1 but not more than X2.

Cumulative probabilities are calculated with a normal approximation if the parameters and the argument are such as to make it accurate. Otherwise the module computes sums of individual terms.

Module 10 (CMODW) Pascal Distribution

This module enables the user to investigate a Pascal distribution after entering the success and process parameters. Options available are the same as for the binomial distribution. Normal approximation or direct summation is used, depending on the values of parameters and the argument.

Module 11 (CMODX) Beta-Binomial Distribution

This module enables the user to study a beta-binomial distribution after specifying the number of observations and parameters of the beta distribution. The same options are available as for the binomial distribution. Probabilities are computed by an interative procedure (Lord & Novick, 1968), with a Stirling approximation for the gamma function.

Module 12 (CMODY) Beta-Pascal Distribution

This module enables the user to examine a beta-Pascal distribution after entering the number of successes and parameters of the beta distribution. Options available are the same as for the binomial distribution.

Module 13 (CMOD55) Poisson Distribution

This module is for study of a Poisson distribution with a specified mean. Options are the same as for the binomial distribution. The CDF is calculated by summing individual terms which are computed by iteration.

Module 14 (CMOD57) Gamma Distribution

The gamma distribution is the same as the chi-square, except that parameters are defined differently (Novick & Jackson, 1974). The module accepts parameters of the gamma distribution, converts them into those of the chi-square distribtuion, and then proceeds as in CMODO.

Module 15 (CMOD9B) Bivariate Normal Distribution

This module allows the user to investigate a bivariate normal distribution after entering the means, standard deviations, and correlation. Following options are available:

1. Probability that X is less than X0 and Y is less than Y0
2. Contour plot of probability density function
3. Highest density regions
4. Conditional distributions
5. Marginal distributions
6. Plot of cumulative distribution (as in Option 1)

This module provides the mean and standard deviation of a conditional or marginal distribution. If the user wants to study the distribution in detail, the module uses CMODE as a subroutine. The module uses NDTR for univariate and BINORM for bivariate normal CDF.

Module 16 (CMOD9L) Multivariate Normal Distribution

This module provides information about a multivariate normal distribution whose means and variance-covariance matrix are given. For the relevant theory see Anderson, 1958. The options are:

1. Highest density regions
2. Univariate conditional distributions
3. Univariate marginal distributions

This module provides the mean and standard deviation of a conditional or marginal distribution. If the user wants to study the distribution in detail, the module uses CMODE as a subroutine.

Module 17 (CMOD9M) Multivariate t Distribution

This module enables the user to study a multivariate t distribution after entering the degrees of freedom, means, and the variance-covariance matrix. Options are the same as for the multivariate normal distribution. For relevant theory see Box and Tiao, 1973.

Module 18 (CMOD9S) Dirichlet (Multivariate Beta) Distribution

This module provides the parameters, mean, standard deviation, and mode of any univariate marginal distribution, which is a two-parameter beta. If more detailed study of the marginal distribution is desired, CMODB is used as a subroutine.

VII. LISTING OF THE CADA PROGRAM MODULES

Cada Modules are made up of a chained sequence of programming elements called CMODs. These are the building blocks from which models are constructed. The function of the supervisory CMODs (e.g., CADA, CEXPLN, CMONTR, and so on) have been explained. (See sections IV and V). The following is a list of the CMODs and a brief description of each module.

CADA	Initialization of the personal file and system parameters
CEXPLN	Explanation of CADA
CMONTR	Control of branching for the model and module levels
CMPGRP	Control of branching for the component and component group levels
RSTRT	Control of branching for restart
CERROR	Handling of unexpected errors
COMP11	Control of the Data Structures component
COMP12	Control of the Data Movement component
COMP13	Control of the Data Transformations component
COMP14	Control of the File Maintenance component
COMP21	Control of the Binary Models component
COMP22	Control of the Univariate Normal Models component
COMP23	Control of the Multicategory Models component
COMP24	Control of the Simple Linear Regression Analysis component
COMP25	Control of the Multiple Linear Regression Analysis component
COMP31	Control of the Utilities and Expected Utilities component
COMP32	Control of the Educational and Employment Selection component
COMP33	Control of the Selection of Educational Treatment component

COMP41	Control of the Simultaneous Estimation of Proportions component
COMP42	Control of the Simultaneous Estimation of Means component
COMP43	Control of the Simultaneous Estimation in M Groups component
COMP51	Control of the Full-Rank Model I Analysis of Variance component
COMP52	Control of the Bayesian Analysis of Repeated-Measures Designs component
COMP61	Control of the Bayesian Full-Rank Multivariate Analysis of Variance component
COMP71	Control of the Frequency Distributions component
COMP72	Control of the Summary Statistics component
COMP73	Control of the Graphic Displays component
COMP74	Control of the Regression component
COMP81	Control of the Univariate Exploratory Data Analysis component
COMP82	Control of the Bivariate Exploratory Data Analysis component
COMP91	Control of the Evaluation of Probability Distributions component
CMOD2.	Beta prior distribution
CMOD3	Beta posterior distribution
CMOD4	Prior distribution for two-parameter-normal standard deviation
CMOD5	Prior distribution for two-parameter-normal mean
CMOD6	Posterior distribution for two-parameter normal
CMOD7	Explanation for comparison of two normal means
CMOD8	Specification of typical prior for m-group proportions, using arcsine transformation
CMOD9	Posterior distribution for m-group proportions, using arcsine transformation

CMOD11	Data entry from the terminal
CMOD12	Data display and editing
CMOD13	Summary statistics
CMOD14	Data graphs and tabular displays
CMOD15	Control for nullary, unary, and binary operations
CMOD16	Set-up module for data grouping
CMOD17	Data transfer to disk
CMOD18	Data transfer from the catalog
CMOD20	Catalog data-set loading
CMOD22	Consensus preposterior analysis for two-parameter normal
CMOD30	Required but missing data set in the personal file
CMOD31	Data transfer from disk
CMOD42	Bivariate plots
CMOD43	Absolute frequency histograms
CMOD55	Evaluation of Poisson distribution
CMOD57	Evaluation of gamma distribution
CMOD61	Working module for data grouping
CMOD62	Equal-slopes model for simultaneous prediction in m-groups
CMOD63	Predictive distribution, equal-slopes model, simultaneous prediction
CMOD64	Adversary preposterior for two-parameter normal
CMOD65	Prior for simple linear regression
CMOD66	Posterior on slope and residual standard deviation
CMOD67	Posterior predictive distribution
CMOD76	Prior distribution for simultaneous estimation of means
CMOD77	Posterior analysis for simultaneous estimation of means

CMOD90	Determination of cut score for one-group selection
CMOD91	Preparation for assessing utilities in two-group restricted selection
CMOD94	Presimultaneous estimation module
CMOD98	Stem-and-leaf for univariate EDA
CMOD99	Box plot for univariate EDA
CMODA	Posterior analysis for m-group proportions using arcsine transformation
CMODB	Evaluation of a beta distribution
CMODC	Evaluation of a Student's t distribution
CMODD	Evaluation of an inverse-chi distribution
CMODE	Evaluation of a normal distribution
CMODF	Evaluation of a Behrens-Fisher distribution
CMODG	Prior for multinomial Dirichlet
CMODH	Posterior multinomial Dirichlet
CMODJ	Entry of indifference probabilities for fixed-state utility assessment
CMODK	Estimation for fixed-state utility assessment
CMODL	Fit of normal and t utility functions
CMODN	Evaluation of an inverse chi-square distribution
CMODO	Evaluation of a chi-square distribution
CMODQ	Explanation for fixed-state utility assessment
CMODR	Control of the evaluation of utility functions
CMODT	Independent beta-distributed proportions
CMODU	Comparison of two standard deviations
CMODV	Evaluation of binomial distribution
CMODW	Evaluation of Pascal distribution

CMODX	Evaluation of a beta-binomial distribution
CMODY	Evaluation of a beta-Pascal distribution
CMODZ	Evaluation of an F distribution
CMOD9A	Empirical CDF for univariate EDA
CMOD9B	Evaluation of bivariate normal distribution
CMOD9C	Least squares and Bayesian estimates, simultaneous estimation
CMOD9D	Selection of variable for univariate EDA
CMOD9E	Selection of analysis for univariate EDA
CMOD9H	Empirical PDF for univariate EDA
CMOD9I	Normal probability plot for univariate EDA
CMOD9J	Smoothed empirical PDF for univariate EDA
CMOD9K	Standardization for univariate EDA
CMOD9L	Evaluation of multivariate normal distribution
CMOD9M	Evaluation of multivariate t distribution
CMOD9N	Retrieval of original Data for univariate EDA
CMOD9O	Transformation of univariate EDA
CMOD9P	Assignment with threshold utility assessment
CMOD9Q	Optional assessments for assignment to educational treatments
CMOD9R	Restricted selection with simple or multiple predictors
CMOD9S	Evaluation of Dirichlet distribution
CMOD9V	Selection of variables for bivariate EDA
CMOD9W	Trimming for univariate EDA
CMOD9Y	Determination of cut score for restricted selection
CMODA1	Selection of analysis for bivariate EDA

CMODA2	Explanation of scatter plot for bivariate EDA
CMODA3	Questions for univariate EDA
CMODA4	Description of univariate EDA
CMODA5	Description of bivariate EDA
CMODA6	Questions for bivariate EDA
CMODA7	Scatter plot for bivariate EDA
CMODA8	Schematic plot explanation for bivariate EDA
CMODA9	Summary statistics for bivariate EDA
CMODAA	Prior for beta-Pascal model
CMODAB	Preposterior for beta-binomial model
CMODAC	Posterior for beta-Pascal model
CMODAI	Schematic plot for bivariate EDA
CMODAJ	Standardization for bivariate EDA
CMODAK	Conditionals for bivariate EDA
CMODAL	Transformation for bivariate EDA
CMODAM	Trimming for bivariate EDA
CMODAN	Retrieval of original data for bivariate EDA
CMODAO	Generalized beta-cumulative utility fit
CMODAP	Retrieval of data after conditional for bivariate EDA
CMODAR	Control and evaluation of least-squares utility functions
CMODAS	Selection of conditional analysis for bivariate EDA
CMODAT	Expected utilities
CMODB1	Least-squares set-up for classical linear regression
CMODB2	Analysis of classical linear regression
CMODB3	Conditional expectation plot
CMODB4	Conditional-utility preliminary assessment

CMODB5	Entry of previously-assessed conditional utilities
CMODB6	Expected utility for conditional utilities
CMODB7	Regional-coherence utility assessment
CMODB8	Local-coherence utility assessment
CMODB9	Explanation and selection of conditional-utility assessment
CMODBA	Explanation and entry for regional-coherence assessment
CMODBB	Explanation and entry for local-coherence assessment
CMODBC	Threshold utility assessment
CMODBD	Control of assignment to treatments
CMODBE	Assessment of utilities for two groups: scaling the two utilities
CMODBF	Two-treatment assignment
CMODBI	Assessment of utilities for two groups
CMODBJ	Assignment to treatments
CMODBK	Selection of applicants, prediction from sample data
CMODBL	Control of restricted selection analysis
CMODBO	Explanation of matrix operations
CMODBP	Standard and special matrix operators
CMODBQ	Matrix management
CMODC1	Explanation of assessment procedure and entry of variables
CMODC2	Estimation of central tendency of the distribution of the regression coefficients
CMODC3	Hypothetical-data method for the estimation of the scale factor and dispersion matrix
CMODC4	Display of the distributions for variance of error and regression coefficients
CMODC5	Selection of variables from sample data and entry of prior information

CMODC6	Computation and display of the posterior distribution for variance of error and regression coefficients
CMODC7	Further examination of the posterior distributions for variance of error and regression coefficients
CMODC8	Display of observed and predicted criterion values for sample data
CMODC9	Storage of parameters from regression equation
CMODCA	Determination of cut scores for each assignment
CMODD0	Determination of design lay-out from raw data for MANOVA
CMODD1	Assembly of summary statistics for ANOVA
CMODD2	Entry of summary statistics from the terminal for ANOVA
CMODD3	Generation of main effects and interactions for ANOVA
CMODD4	Tutorial on main effects and interactions for ANOVA
CMODD5	Analysis of posterior t-distributions of effects for ANOVA
CMODD6	Calculation of probability content of smallest highest density region (HDR) for ANOVA
CMODD7	Tutorial on highest density regions (HDRs) for ANOVA
CMODDA	Entry of summary statistics or priors parameters from terminal for MANOVA
CMODDB	Compute parameters of posterior distribution for MANOVA
CMODDC	Definitions of main effects for MANOVA
CMODDD	Examination of the posterior matric t distribution for MANOVA
CMODDE	Continuation from CMODD0: compute summary statistics for MANOVA
CMODDF	Continuation from CMODD0: collect lay-out design from terminal for MANOVA
CMODDG	Continuation from CMODDC: compute transformation matrix for effects for MANOVA
CMODDH	Continuation from CMODDD: collect parameter statistics for MANOVA

CMODDI	Continuation from CMODDD: sweep operator for MANOVA
CMODDJ	Continuation from CMODDD: beta distribution, approximate content of HDR for MANOVA
CMODDK	Tutorial I for MANOVA
CMODDL	Tutorial II for MANOVA
CMODDM	Storage of parameters for ANOVA and MANOVA
CMODE0	Explanation for nullary, unary, and binary operations
CMODE1	Operation definition for nullary, unary, and binary operations
CMODE2	Operation execution for nullary, unary and binary operations

VIII. LISTING OF CADA PROGRAM MODULES AND DATA FILES REQUIRED FOR EACH COMPONENT GROUP

Many CADA users have only small disks available on their computers and therefore can not store all of CADA either in memory or on disk. Because the number of CADA users having access to such computers is increasing, CADA has been segmented into component groups. This has been done in such a way that usually only the control programs and one component group need be loaded at one time.

In this section, we list the modules needed for each of the component groups. Users who wish to load only a portion of CADA should follow the special loading instructions provided in section XI and the instructions provided with CADA, using the information given here as to what modules are needed for each component group.

Component Group 1

Data Management Facility

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP11
	COMP12	COMP13	COMP14	RSTRT	
Component 11.	Data Structures (not yet implemented)				
Component 12.	Data Movement				
	CMOD11	CMOD12	CMOD17	CMOD18	CMOD20
	CMOD31				
Component 13.	Data Transformations				
	*CMOD13	CMOD15	CMODBO	CMODBP	CMODBQ
	CMODEO	CMODE1	CMODE2		
Component 14.	File Maintenance				
	CMOD16	CMOD61			

*Also accessible from component groups 7 and 8.

-- Data Files Needed --

A0, A1, ... (Personal Files)

FILE1	FILE2	FILE3	FILE4	FILE5
FILE6	FILE7	C8LL6	C8LL7	C8LL8
C8LL9	C8LL10	C8LL11	C8LL12	C8LL13
C8LL15	C8LL19	C9LL6	C9LL7	C9LL8
C9LL9	C9LL10	C9LL11	C9LL12	C9LL13
C9LL15	C9LL19	DFIL01	DFIL02	. . .

Component Group 2

Simple Bayesian Parametric Models

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP21
	COMP22	COMP23	COMP24	COMP25	RSTRT
Component 21.	Binary Models				
	CMOD2	CMOD3	CMODAA	CMODAB	CMODAC
	*CMODB	CMODT	*CMODX	*CMODY	
Component 22.	Univariate Normal Models				
	CMOD4	CMOD5	CMOD6	CMOD7	CMOD22
	CMOD64	*CMODC	*CMODD	*CMODF	CMODU
Component 23.	Multicategorical Models				
	CMODG	CMODH			
Component 24.	Simple Linear Regression Analysis				
	CMOD65	CMOD66	CMOD67	CMODC9	
Component 25.	Multiple Linear Regression Analysis				
	CMODC1	CMODC2	CMODC3	CMODC4	CMODC5
	CMODC6	CMODC7	CMODC8	CMODC9	

*Also accessible from Component Group 9.

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 3 *

Decision Theoretic Models

--Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP31
	COMP32	COMP33	RSTRT		
Component 31.	Utilities and Expected Utilities				
	CMODJ	CMODK	CMODL	CMODQ	CMODR
	CMODAO	CMODAR	CMODAT	CMODBA	CMODBB
	CMODB7	CMODB8			
Component 32.	Educational and Employment Selection				
	CMOD90	CMOD91	CMOD9R	CMOD9Y	CMODBC
	CMODBE	CMODBI	CMODB[illegible]	CMODBL	
Component 33.	Educational and Employment Assignment				
	CMOD9P	CMOD9Q	CMODBD	CMODBF	CMODBJ
	CMODB4	CMODB5	CMODB6	CMODB7	CMODB8
	CMODB9	CMODCA			

*Component Group 2 must be loaded to use this component group.

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 4

Bayesian Simultaneous Estimation

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP41
	COMP42	COMP43	RSTRT		
Component 41.	Simultaneous Estimation of Proportions				
	CMOD8	CMOD9	CMODA		
Component 42.	Simultaneous Estimation of Means				
	CMOD76	CMOD77			
Component 43.	Simultaneous Estimation in m-Groups				
	CMOD62	CMOD63	CMOD94	CMOD9C	

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 5

Bayesian Full-Rank ANOVA

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP51
	COMP52	RSTRT			
Component 51.	Full-Rank Model I Analysis of Variance				
	CMODD1	CMODD2	CMODD3	CMODD4	CMODD5
	CMODD6	CMODD7	CMODDM		
*Component 52.	Bayesian Analysis of Repeated-Measures Designs				
	CMODDO	CMODDA	CMODDB	CMODDC	CMODDD
	CMODDE	CMODDF	CMODDG	CMODDH	CMODDI
	CMODDJ	CMODDK	CMODDL	CMODDM	

*Component 52 is essentially the same as Component 61.

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 6

Bayesian Full-Rank Multivariate Analysis

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP61
	RSTRT				
*Component 61.	Bayesian Full-Rank Multivariate Analysis of Variance				
	CMODD0	CMODDA	CMODDB	CMODDC	CMODDD
	CMODDE	CMODDF	CMODDG	CMODDH	CMODDI
	CMODDJ	CMODDK	CMODDL	CMODDM	

*Component 61 is essentially the same as Component 52.

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 7

Classical Elementary Statistics

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP71
	COMP72	COMP73	COMP74	RSTRT	
Component 71.	Frequency Distributions				
	CMOD14	CMOD43			
Component 72.	Summary Statistics				
	*CMOD13				
Component 73.	Graphic Displays				
	CMOD42	CMOD43			
Component 74.	Regression				
	CMODB1	CMODB2	*CMOD15	CMOD91	CMOD42

*Also accessible in Component Groups 1 or 8.

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 8

Exploratory Data Analysis

-- Modules Needed --

Control	CERROR	*CMOD30	CMONTR	CMPGRP	COMP81
	COMP82	RSTRT			
Component 81.	Univariate Exploratory Data Analysis				
	*CMOD13	CMOD98	CMOD99	CMOD9A	CMOD9D
	CMOD9E	CMOD9H	CMOD9I	CMOD9J	CMOD9K
	CMOD9N	CMOD9O	CMOD9W	CMODA3	CMODA4
Component 82.	Bivariate Exploratory Data Analysis				
	*CMOD13	CMOD98	CMOD99	CMOD9A	CMOD9H
	CMOD9I	CMOD9J	CMOD9V	CMODA1	CMODA2
	CMODA5	CMODA6	CMODA7	CMODA8	CMODA9
	CMODAI	CMODAJ	CMODAK	CMODAL	CMODAM
	CMODAN	CMODAP	CMODAS	CMODB3	

*Also accessible in Component Groups 1 and 7.

-- Data Files Needed --

A0, A1, ... (Personal Files)

Component Group 9

Probability Distributions

-- Modules Needed --

Control	CERROR	CMOD30	CMONTR	CMPGRP	COMP91
	RSTRT				

Component 91. Evaluation of Probability Distributions

CMOD55	CMOD57	*CMODB	*CMODC	*CMODD
CMODE	*CMODF	CMODN	CMODO	CMODV
CMODW	*CMODX	*CMODY	CMODZ	CMOD9B
CMOD9L	CMOD9M	CMOD9S		

*Also accessible in Component Group 2.

-- Data Files Needed --

A0, A1, ... (Personal Files)

IX. COMPUTATIONAL SUBROUTINES

Since many mathematical manipulations are the same from module to module, CADA's data analysis routines have been built in a subroutine manner. A large percentage of the computation done by CADA involves the use of 16 computational subroutines. The following is a list of these 16 subroutines, together with a brief description of the computational methods they employ.

Subroutine	Description
BDTR	Beta cumulative distribution function (CDF)
BHDR	Beta highest density region (HDR)
BINORM	Bivariate normal cumulative distribution function (CDF)
BISORT	Sort bivariate data
CSQDTR	Chi-square cumulative distribution function (CDF)
CSQHDR	Chi-square highest density region (HDR)
ICHDR	Inverse-chi highest density region (HDR)
ICSQHR	Inverse-chi-square highest density region (HDR)
INPCHR	Character input from the terminal
INPNUM	Numeric input from the terminal
LGAM	Natural logarithm of the gamma function
NDTR	Normal cumulative distribution function (CDF)
PFOPEN	Open the personal file
SORT	Sort univariate data
TDTR	Student's t cumulative distribution function (CDF)
THDR	Student's t highest density region (HDR)

BDTR

This subroutine calculates the cumulative distribution function (CDF) of a beta distribution. Arguments consist of parameters A and B of the distribution, and the value J2 where the CDF is desired. The result is returned in variable P.

If A or B is less than 5, the distribution is assumed to be approximately normal when (A+B) exceeds 85; otherwise 16-point Gaussian integration is carried out. Peizer-Pratt approximation is used when both A and B are at least 5 (Peizer & Pratt, 1968).

A GOSUB to initialize the Gaussian constants is needed once before this subroutine is called for the first time. After the parameters A and B are entered, a GOSUB is needed to calculate the normalizing constant F0 for the beta distribution. F0 is the log of the reciprocal of the complete beta function.

Arrays used are O and W which contain the points and weights for Gaussian integration. Scalars used are A, B, C6, C7, D0, D1, D9, F0, G0, G9, J2, P, R0. Subroutines called are NDTR for normal CDF and LGAM for log gamma.

BHDR

This subroutine calculates highest density regions for a beta distribution. Arguments consist of parameters A and B of the distribution, and the probability J5 contained in the HDR. Both A and B must exceed 1 for an HDR to exist. The lower and upper end points of the region are returned in J1 and J2.

The logic is based on the formula given by Jackson (1974) for J1 and J2 as functions of the ratio d = J1/J2. Let μ and σ be the mean and standard deviation of the distribution. The initial value for d is $(\mu+1.6\sigma J5)/(\mu-1.6\sigma J5)$. J1, J2 and the probability between them are calculated. The Newton-Raphson procedure is used to improve the value of d until the probability between J1 and J2 is within .0001 of J5.

Scalars used are A, A9, B, C, J, J1, J2, J3, J4, J5, J6, J7, J8, P, P0, P9, U0. Subroutines needed are BDTR for beta CDF, NDTR for normal CDF, and LGAM for log gamma.

BINORM

This subroutine computes the probability in the region $(X<H, Y<K)$ of a standard bivariate normal distribution. Arguments consist of the correlation R and the values H, K. The probability is returned in S.

Probability is calculated by summing the first 20 terms of the tetrachoric series of Pearson (1900).

Arrays used are T, V, W. Scalars used are H, H1, I, K, K1, N, 06, 07, 08, P, P2, Q3, R, R9, S. Subroutine NDTR is needed for calculating univariate normal CDF.

BISORT

This routine sorts bivariate observations stored in two-dimensional array P into ascending order based on the values of the observations in the first column. This is a destructive sort in that the original data are not saved. The number of observations must be set in variable NO. This subroutine makes use of a transposition bubble sort-technique (Nievergelt, Farrar, and Reingold, 1974). The array P is used, as are scalars I, N, NO, and R0 through R9.

CSQDTR

This subroutine provides the cumulative distribution function of a standard chi-square distribution. Arguments are G, the degrees of freedom, and X, the point at which CDF is desired. The result is returned in P.

Probability is calculated using the normal approximation of Peizer and Pratt, 1968.

Variables used are B, B3, G, G6, P, P2, T, X, X2. Subroutine NDTR is needed for the normal CDF.

CSQHDR

This subroutine computes highest density regions for a standard chi-square distribution. Arguments consist of the degrees of freedom G and the probability J5 in the HDR. The lower and upper end points are returned in J1 and J2.

For any given length of the HDR, its end points are calculated using the formulae of Jackson (1974). Probability between these points is calculated and compared with J5. The iterative procedure recommended by Jackson is followed until the probability is within .00001 of J5.

Variables used are G, J, J0, J1, J2, J3, J5, J8, J9, X. Subroutines called are CSQDTR for chi-square probabilities and NDTR for normal CDF.

ICHDR

This subroutine calculates highest density regions for a standard inverse chi distribution. Arguments are the degrees of freedom G and the probability J5. The results consist of the lower and upper end points J1 and J2.

The logic of the algorithm is the same as in CSQHDR.

Variables are J, J0, J1, J2, J3, J8, J9, P, X. Subroutines needed are CSQDTR and NDTR.

ICSQHR

This subroutine provides highest density regions of a standard inverse chi-square distribution. Arguments consist of G, the degrees of freedom, and J5, the probability in the HDR. The lower and upper end points of the interval are returned in J1 and J2.

Logic is the same as in CSQHDR.

Variables used are J, J0, J1, J2, J3, J8, J9, P, X. Subroutines called are CSQDTR and NDTR.

INPCHR

This routine allows the user to enter a character string up to 72 characters long. The calling sequence is:

```
GOSUB 9050
```

The string is returned in the variable F$. If the character string "-9999" is entered, then RSTRT is called, giving the user the opportunity to restart.

INPNUM

This routine allows the user to enter up to five numbers from the terminal. The calling sequence is:

```
Let 00 = n
GOSUB 9000
```

where n (1<= n <= 5) is the number of values desired. The values are returned in 01, 02, 03, 04, 05, depending on how many are requested. If any of the values are equal to -9999, the RSTRT is called, giving the user the opportunity to restart.

LGAM

This subroutine calculates the natural logarithm of the gamma function at any (positive) values of the argument G9. The result is returned in GØ.

If G9 exceeds 18, a Stirling approximation for log of a factorial is used, with log (G9) subtracted. (For values of the coefficients see, for example, Dahlquist, Bjorck & Anderson, 1974, p. 301). For smaller values of G9, the quantity used in the Stirling approximation is G5 which exceeds 18 and differs from G9 by an integer. Then the log of G9*(G9+1)*...*(G5-1) is subtracted.

Variables used are C1, C2, C3, C4, G0, G5, G6, G9, R8.

NDTR

This subroutine provides the cumulative distribution function of a standard normal distribution. The argument is N3 and the result is P.

The algorithm is a rational approximation from Hastings (1955, p. 169).

Variables used are C1, C2, C3, C4, C5, D, N3, N4, P, T, X, X1.

PFOPEN

This routine opens the personal file and reads three numbers identifying the module, model, and component. It also rewrites the record if desired. There are three calling sequences:

1. Open the personal file and read three values from record 1.

```
J3 = n
J2 = n
J1 = n
GOSUB 9060
```

   where n is -1 if the user desires the corresponding value in the personal file unchanged, and zero or greater if the user desires that the value in the personal file can be changed. The values found in the personal file are returned in I3, I2, and I1.

2. Read three values from the personal file, assumed to have been previously opened, and rewrite the values if desired.

J3 = n
J2 = n
J1 = n
GOSUB 9067

where n, J3, J2, and J1 hve the same meanings as before. Values are returned in I3, I2, and I1.

3. Write up to three new values in the personal file.

J3 = n
J2 = n
J1 = n
GOSUB 9068

where J3, J2, J1 and n have the same meanings as before. No values are returned; I3, I2, and I1 are left unchanged.

<u>SORT</u>

This subroutine sorts univariate observations stored in vector Q into ascending order. This is a destructive sort since the original values of Q are destroyed. The number of observations must be set in variable NO. This subroutine makes use of a transposition bubble-sort technique (Nievergelt, Farrar, and Reingold, 1974). The array Q is used, as are scalars I, J, NO, and R0 through R4.

<u>TDTR</u>

This subroutine calculates the cumulative distribution function for a standard t distribution with G degrees of freedom, at t = J2. The CDF is returned in P.

If G = 1, the arctangent function is used (Abramowitz & Stegun, 1964, eqn. 26.7.3). Sixteen-point Gaussian quadrature is performed if G is between 1 and 10. The approximation given by Peizer and Pratt (1968) is used if G is 10 or above. P = 1 if ($G \geq 6$, $J2 \geq 6$) or ($G < 6$, $J2 > 12$).

Before the first call to this subroutine, a GOSUB is needed to initialize the constants for Gaussian integration. Arrays used are O and W. Scalars are D0, D1, D9, F0, G, G0, G9, I9, J1, J2, N, N1, P. Subroutines needed are NDTR and LGAM.

THDR

This subroutine provides the highest density regions of a standard student's t distribution. Arguments consist of G, the degrees of freedom, and J5, the probability content of the HDR. The upper end point of the interval is returned in J2; the interval is symmetric about zero. The logic is the same as in CSQHDR.

Variables used are G, J1, J2, J5, N, P, X0, X2, X9, Z8, and Z9; subroutine TDTR is called.

X. DATA FILES

The CADA Monitor makes extensive use of data files. Each user of the Monitor is temporarily assigned a unique personal file in which the data he or she creates are saved during the course of a CADA session. This file gives the user the capability to move from module to module, model to model, or component to component, without losing pertinent data. However, when the user finishes his or her current run of CADA, these data are lost.

Another type of data file associated with CADA is the permanent data file. There are 36 permanent data files provided with the Monitor. Of these, 27 contain data sets that are accessed from Component 12, model 1, module 3, Data File Catalog, and were previously explained. The other nine data files are for storage of user data-sets. These have passwords associated with them and can only be accessed by a user with the correct password. These nine files are: DFIL01, DFIL02, DFIL03, DFIL04, DFIL05, DFIL06, DFIL07, DFIL08, DFIL09. The passwords are the first entry in each file and consist of a six-character string. A list of the data files is in Table 2. Reading and writing data to and from files are highly system-dependent. Although CADA has the facilities on our development system for reading and writing data to external files, this capability may not be implemented on all systems. CADA expects the following format for all of the above files:

P$ = password (6 characters)

N$ = data-set name (6 characters)

GO = number of groups (0 if not grouped data, maximum is 12)

VO = number of data elements (maximum is 5)

G$ = group names (72-characters maximum, 6 characters per name)

V$ = data element names (30-characters maximum, 6 characters per name

N = group counts (12 numbers, one for each possible group; zeros for unused groups)

D = data (maximum 1000 entries stored by variable within observation within group)

The file is assumed to be written by a Basic program.

TABLE 2

C8LL6	ACT - 1968 - School 6	C9LL6	ACT - 1969 - School 6
C8LL7	ACT - 1968 - School 7	C9LL7	ACT - 1969 - School 7
C8LL8	ACT - 1968 - School 8	C9LL8	ACT - 1969 - School 8
C8LL9	ACT - 1968 - School 9	C9LL9	ACT - 1969 - School 9
C8LL10	ACT - 1968 - School 10	C9LL10	ACT - 1969 - School 10
C8LL11	ACT - 1968 - School 11	C9LL11	ACT - 1969 - School 11
C8LL12	ACT - 1968 - School 12	C9LL12	ACT - 1969 - School 12
C8LL13	ACT - 1968 - School 13	C9LL13	ACT - 1969 - School 13
C8LL15	ACT - 1968 - School 15	C9LL15	ACT - 1969 - School 15
C8LL19	ACT - 1968 - School 19	C9LL19	ACT - 1969 - School 19
FILE1	ITBS, School 1	DFIL01	User data
FILE2	ITBS, School 14	DFIL02	User data
FILE3	ESAA Pilot Program Data	DFIL03	User data
FILE4	Iowa County Data	DFIL04	User data
FILE5	Sample Regression Data	DFIL05	User data
FILE6	Sample ANOVA Data	DFIL06	User data
FILE7	Sample MANOVA Data	DFIL07	User data
		DFIL08	User data
		DFIL09	User data

There are a number of ways in which data may be made available to CADA. First, users may enter data at their terminals via CADA's data entry from terminal facility (component 12, model 1, module 1). Second, users may add data files to the data file catalog. This will entail modifying CMOD18 to accept additional data files. Two more data files, which must be named "FILE8" or "FILE9", may be added easily. CMOD18 must then be modified by adding to the menu of data files, by adding additional description or descriptions, and by changing the check for legal option selection from seven to eight or nine. If you wish to add more than two catalog data files, CMOD20 must be modified as well.

A third way to make data available to CADA is to use the nine user data files (DFIL01, ..., DFILE09). You may place your data in a data file in the format described above, name that data file "DFILxx" (where 'xx' is 01, ..., or 09) and use CADA's facilities for retrieving data from user-stored disk files (compoent 12, model 1, module 2). The DFILs must be stored with the rest of the CADA system. In addition, you may add the capacity to handle more than nine DFILs to CADA. The new DFILs should be named DFIL10 through DFILnn, where nn <=99; they should be consecutively numbered. You must change "DO" in CMOD17 and CMOD31 to reflect the number of DFILs allowed. This number is now nine.

If you create a new DFIL, or wish to change the password of an existing one, the file must be initialized. CADA expects at least a password (maximum of 6 characters) and a blank data-set name (maximum of 6 characters). These two strings must be written by a Basic program.

XI INSTRUCTIONS FOR INSTALLING THE CADA MONITOR

Since many users of CADA have limited mass storage, CADA has been divided into nine component groups. It is only necessary to load the modules for the desired component group (and in some instances Component Group 2 or Component Group 9; see section VIII for listing of modules and data files for each component group). This allows the user to use a relatively small amount of on-line disk space for CADA. A large portion of CADA may be stored on an off-line medium such as magnetic tape and a component group loaded to disk only when it is needed. However, this does not preclude all of CADA being kept on-line, if desired.

CADA is distributed on the storage medium that we feel is best suited for each computer system. This is magnetic tape for all systems for which CADA-1980 is supported. The characteristics of data transfer and storage are different for each system and are not listed here. Accompanying each order are instructions for loading CADA to the specified system. These instructions list the characteristics of the medium and the method for loading CADA that is the easiest to use. CADA will often be loaded by a utility program provided by the manufacturer of the user's computer system. However, not all manufacturers provide this facility. If a program is not provided by the manufacturer, a set of instructions for loading CADA is listed.

XII. INSTRUCTIONS FOR USING THE CADA MONITOR

There are only three system commands that the investigator need know to use this package. These are: a sign-on command, a command to start the CADA system, and a sign-off command. The sign-on procedure varies from system to system. Once the investigator is signed on the system, the initial module must be loaded and executed, which is usually accomplished by one command. The name of the module to be executed is CADA. On the HP 2000 series, the load and execute command is "EXEC-CADA". When finished with the system, the user must sign off, usually by typing the command "BYE".

In addition to the three basic commands, the user should know how to use the restart facility in CADA. CADA has a built-in restart capability so that if a user wishes to terminate an analysis, he or she need only type "-9999" when asked for input. Any necessary data are retained in the user's personal file. The user is then given the following options:

1. start at the beginning of the module in which he or she is now operating (without loss of data in the personal file);
2. begin a new module within the current model;
3. select a new model within the current component;
4. choose a new component within the current component group;
5. change to a new component group;
6. exit the CADA Monitor.

XIII. DESCRIPTION OF USER PERSONAL FILES

The length of the personal file assigned to each user is twenty-two 512-byte records in the HP 2000 system implementation. The beginning of each record is accessed randomly, but data is accessed sequentially within the record. The records used as entry points are listed in Table 3.

TABLE 3

Records (Entry Points)	Sequential File Names
1	RFIL1
2	RFIL2
3	RFIL3
4	RFIL4
5	RFIL5
7	RFIL7
12	RFIL12
14	RFIL14
15	RFIL15

Across from each record number is the name of the file used for the record in those systems that allow only sequential access. On systems that do not have the ability to pass information to a chained module through common memory, RFILE0 is used to pass the terminal screen-erase characters.

The personal files for the HP 2000 implementation of CADA are named A0, A1, A2, A3, A4, A5, A6, A7, A8, A9, AA, AB, AC, AD, AE, and AF. This allows up to 16 simultaneous users. If you wish fewer simultaneous users, delete the unwanted files starting with AF and deleting toward A0. If at most three simultaneous users are desired, then only files A0, A1, and A2 need to be stored. On systems that may have files local (i.e., unique) to the terminal session, a personal file will be created locally and should be purged when the terminal session is complete.

XIV. THE CADA DISPLAY BOOK

The CADA Display Book, a companion but separate publication, is available. It provides examples of the use of each component in the Monitor. Although it is not feasible to give an example of every possible option available in the Monitor, the Display Book does show examples for most of the modules in all of the models of each component.

The examples in the Display Book were used to test each translation. It is advised that the CADA system administrator use the examples in the Display Book to check the implementation on the local computer system.

The Display Book may also be used by new users to become familiar with the Monitor. Once the user has seen how a particular analysis is presented, other options than those of the example analysis may be chosen to see how the results obtained may vary. An order form is included at the end of this manual.

XV. REFERENCES

Abramowitz, M., & Stegun, I. A. Handbook of mathematical functions. U.S. Department of Commerce, NSB Applied Mathematical Series, 1964.

American National Standards Institute Technical Committee X3J2. Proposed American National Standard Programming Language Minimal BASIC. BSR X3.60, X3J2/76-01, 76-01-35.

Anderson, T. W. An introduction to multivariate statistical analysis. New York: Wiley, 1958.

Bargmann, R. E., & Ghosh, S. P. Statistical distribution programs for a computer language. Research Report, RC-1094. Yorktown Heights, New York: IBM Watson Research Center, 1963.

Bell, M. A. The affects of substantive and affective conflict in problem solving groups. Speech Monographs, 1974, 41, 19-24.

Box, G. E. P., & Tiao, G. C. Bayesian inference in statistical analysis. Reading, Massachusetts: Addison-Wesley, 1973.

Dahlquist, G.; Bjorck, A.; & Anderson, N. Numerical Methods. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1974.

Everitt, P. F. Tables of the tetrachoric functions for fourfold correlation tables. Biometrika, 1910, 7, 56.

Froberg, C. E. Introduction to numerical analysis. Reading, Massachusetts: Addison-Wesley, 1965.

Gnanadesikan, R. Methods for statistical data analysis of multivariate observations. New York: Wiley, 1977.

Hastings, C. Approximations for digital computers. Princeton, New Jersey: Princeton University Press, 1955.

IBM System 136 Scientific Subroutine Package Version III (360A-CM-03X). Programmer's manual, fifth edition, Gh20-0205-4. White Plains, New York, 1968.

Isaacs, G. L. Conversational exploratory data analysis. Unpublished doctoral dissertation, The University of Iowa, 1978.

Isaacs, G. L. Interdialect translatability of the BASIC programming language. SIGCUE Bulletin, 1974, 9, 11-22. Has also appeared as ACT Technical Bulletin, No. 11 (reprinted and distributed by 1975). An earlier version was distributed by CONDUIT in 1972.

Isaacs, G. L. BASIC revisited: an update to "interdialect translatability of the BASIC programming language". Iowa City, Iowa: CONDUIT, 1976.

Isaacs, G. L.; Christ, D. E.; Novick, M. R.; & Jackson, P. H. Tables for Bayesian statisticians. Iowa City, Iowa: The University of Iowa, 1974.

Isaacs, G. L., DeKeyrel, D. F., & Novick, M. R. CADA user's manual--1976. Iowa City: Iowa Testing Programs, 1976.

Isaacs, G. L., & Novick, M. R. The Bayesian computer-assisted data analysis (CADA) monitor. Proceedings of ACM SIGSCE-SIGCUE Joint Symposium. New York: Association for Computing Machinery, 1976.

Jackson, P. H. Formulae for generating highest density credibility intervals. ACT Technical Bulletin No. 20. Iowa City, Iowa: The American College Testing Program, 1974.

Jackson, P. H.; Novick, M. R.; & DeKeyrel, D. F. Adversary preposterior analysis for simple parametric models, in A. Zellner (Ed.), Bayesian Analysis in Econometrics and Statistics - Essays in Honor of Harold Jeffreys. Studies in Bayesian Econometrics, Vol. 1. Amsterdam: North Holland Publishing Company, 1980.

Johnson, N. L., & Leone, F. C. Statistics and Experimental Design in Engineering and the Physical Sciences, Volume II. New York: John Wiley & Sons, Inc., 1974.

Kadane, Joseph B.; Dickey, James M.; Winkler, Robert L.; Smith, Wayne S.; and Peters, Stephen C. Interactive Elicitation of Opinion for a Normal Linear Model. Technical Report No. 150, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1979.

Lewis, C. A note on Bayesian simultaneous linear regression with constant slopes. ONR Technical Report 78-1, June, 1978.

Lewis, C.; Wang, M. M.; & Novick, M. R. Marginal distributions for the estimation of proportions in m-groups. Psychometrika, 1975, 40, 63-76.

Lindley, D. V., & Novick, M. R. The role of exchangeability in inference. Annals of Statistics, 1980, in press.

Lindley, D. V., & Smith, A. F. M. Bayes estimates for the linear model. Journal of the Royal Statistical Society (Series B), 1972, 34, 1-41.

Molenaar, W. Approximations to the Poisson, binomial, and hypergeometric distribution functions. Amsterdam, Holland: Mathematical Centre, 1970.

Mosteller, F., & Tukey, J. W. Data analysis and regression. Reading, Massachusetts: Addison-Wesley, 1977.

Neter, J., & Wasserman, W. Applied linear statistical models. Homewood, Illinois: Irwin, 1974.

Nievergelt, J.; Farrar, J. C.; & Reingold, E. M. Computer approaches to mathematical problems. New Jersey: Prentice-Hall, 1974.

Novick, M. R. High school attainment: an example of a computer-assisted Bayesian approach to data analysis. International Statistical Review, 1973, 41, 264-271.

Novick, M. R. A course in Bayesian statistics. The American Statistician, 1975, 29, 94-97.

Novick, M. R.; Chuang, D.; & DeKeyrel, D. F. Local and regional coherence utility assessment procedures. Trabajos de Estadistica, (in press).

Novick, M. R.; Hamer, R. M.; and Chen, J. C. The Computer-Assisted Data Analysis (CADA) Monitor, 1978. The American Statistician, 1979, 33, 219-220.

Novick, M. R.; Isaacs, G. L.; & DeKeyrel, D. F. Computer-assisted data analysis, 1978: manual for the computer-assisted data analysis (CADA) monitor. Iowa City: Iowa Testing Programs, 1977.

Novick, M. R.; Isaacs, G. L.; & DeKeyrel, D. F. Modularity, readability, and transportability of the computer-assisted data analysis (CADA) monitor. Proceedings of the computer science and statistics: eleventh annual symposium for the INTERFACE. Raleigh, North Carolina: Institute of Statistics, North Carolina State University, 1978.

Novick, M. R., & Jackson, P. H. Statistical methods for educational and psychological research. New York: McGraw-Hill, 1974.

Novick, M. R.; Jackson, P. H.; & Thayer, D. T. Bayesian inference and the classical test theory model: reliability and true scores. Psychometrika, 1971, 36, 261-288.

Novick, M. R.; Jackson, P. H.; Thayer, D. T.; & Cole, N. S. Estimating multiple regressions in m-groups: a cross-validation study. The British Journal of Mathematical and Statistical Psychology, 1972, 25, 33-50.

Novick, M. R., & Lewis, C. Prescribing test length for criterion-referenced measurement. CSE Monograph Series in Evaluation, No. 3. Los Angeles: University of California Center for the Study of Evaluation, 1974.

Novick, M. R., & Lindley, D. V. The use of more realistic utility functions in educational applications. Journal of Educational Measurement, 1978, 15, 181-191.

Novick, M. R. & Lindley, D. V. Fixed-state assessment of utility functions. The Journal of the American Statistical Association, 1979, 74, 306-311.

Novick, M. R., & Petersen, N. S. Towards equalizing educational and employment opportunity. Journal of Educational Measurement, 1976, 13, 77-88.

Patil, V. H. The Behrens-Fisher problem and its Bayesian solution. Journal of the Indian Statistical Association, 1964, 2, 21-31.

Pearson, K. Mathematical contributions to the theory of evolution VII: On the correlation of characters not quantitatively measurable. Philosophical Transactions of the Royal Society, Series A, 1900, 195, 1-47.

Peizer, D. B., & Pratt, J. W. A normal approximation of binomial, F, beta and other common, related tail probabilities. Journal of the American Statistical Association, 1968, 63, 1417-1456 (Part I) and 1457-1483 (Part II by J. W. Pratt alone).

Petersen, N. S. An expected utility model for "optimal" selection. Journal of Educational Statistics, 1976, 1, 333-358.

Petersen, N. S., & Novick, M. R. An evaluation of some models for culture-fair selection. Journal of Educational Measurement, 1976, 13, 3-29.

Pratt, J. W.; Raiffa, H.; & Schlaifer, R. O. Introduction to Statistical Decision Theory, New York: McGraw-Hill, 1965.

Raiffa, H., & Schlaifer, R. Applied statistical decision theory. Cambridge, Massachusetts: The Massachusetts Institute of Technology Press, 1961.

Shigemasu, K. Development and validation of a simplified m-group regression model. Journal of Educational Statistics, 1976, 1, 157-180.

Stroud, A. H., & Secrest, D. Gaussian quadrature formulas. Englewood Cliffs, New Jersey: Prentice-Hall, 1966.

Thorndike, R. L. Educational measurement (2nd ed.). Washington, D. C.: American Council on Education, 1971.

Tukey, J. W. Exploratory data analysis. Reading, Massachusetts: Addison Wesley, 1977.

Wang, M. M.; Novick, M. R.; Isaacs, G. L.; & Ozenne, D. A Bayesian data analysis system for the evaluation of social programs. Journal of the American Statistical Association, 1977, 72, 711-712.

Winer, B. J. Statistical Principles in Experimental Design. (Second Edition) New York: McGraw-Hill, 1971.

Woodworth, G. G. t for two, or preposterior analysis for two decision-makers' interval estimates for the mean. The American Statistician, 1976, 30, 168-171.

Woodworth, G. G. Bayesian full rank MANOVA/MANCOVA: An Intermediate Exposition with interactive computer examples. Journal of Educational Statistics, 1979, 4, 357-404.

CADA RESEARCH GROUP

ERROR REPORT

Date ________________ Reporter ________________ Computer ________________

Module name ____

Line number ____

Error Message ____

Describe the sequence of events leading to the error:

Can you reproduce the error? ________________________________

Remarks:

Error corrected by ____________________________ Date __________

Summary of correction: