NSWC TR 89-97

STATLIB: NSWC LIBRARY OF STATISTICAL PROGRAMS AND SUBROUTINES

BY
MARLIN A. THOMAS GARY W. GEMMILL JOHN R. CRIGLER
ENGINEERING & INFORMATION SYSTEMS DEPARTMENT

AUGUST 1989

NAVAL SURFACE WARFARE CENTER
Dahlgren, Virginia 22448-5000 • Silver Spring, Maryland 20903-5000

NSWC TR 89-97

STATLIB: NSWC LIBRARY OF STATISTICAL PROGRAMS AND SUBROUTINES

BY

MARLIN A. THOMAS
GARY W. GEMMILL
JOHN R. CRIGLER

ENGINEERING & INFORMATION SYSTEMS DEPARTMENT

AUGUST 1989

NAVAL SURFACE WARFARE CENTER
Dahlgren, Virginia 22448-5000 • Silver Spring, Maryland 20903-5000

FOREWORD

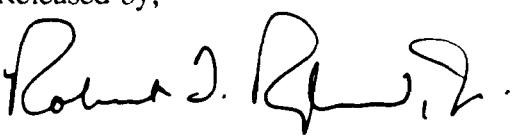
The work documented in this report was performed by the Mathematical Statistics Staff (E406). It was funded by the Computing Systems and Networks Division (E40) and the Space and Surface Systems Division (K10).

The authors gratefully acknowledge the contributions by Mr. Gary M. Johnson of the Mathematical Statistics Staff and by past members of the staff: Ms. Audrey E. Taub of Mitre Corporation, Ms. Patricia A. Shields of the Defense Logistics Agency and the late Mr. Gary J. Sutter. They also express appreciation to Ms. Elissa A. Zizzi for converting the older programs in STATLIB from the CDC 6700 to the CDC 865 and to Ms. J. Diane Bell for setting up the system library procedures for STATLIB.

The authors acknowledge the extensive use of MATHLIB, NSWC's library of numerical mathematics subroutines. They express their gratitude to Alfred H. Morris, Jr., developer of MATHLIB and the author of ***NSWC LIBRARY OF MATHEMATICS SUBROUTINES*** (NSWC TR 86-251) and to Dr. Armido R. DiDonato who formulated many of the algorithms therein.

This report was reviewed by Raymond O. Brancolini, Head of the Computing Systems and Networks Division and J. Ralph Fallin, Head of the Space and Surface Systems Division.

Released by,



ROBERT T. RYLAND, Jr., Head
Engineering & Information Systems
Department

TABLE OF CONTENTS

INTRODUCTION		1
Overview of STATLIB		1
Origin of STATLIB		1
Establishment of STATLIB		1
Commercial Statistical Packages at NSWC		2
USING STATLIB		5
Library Organization		5
How to Call It		5
Information Needed to Run It		5
EXAMPLES		9
Program		9
Subroutine		10
DESCRIPTIONS AND INPUT GUIDES		21
PROGRAMS		23
REGRESSION ANALYSIS		25
GEMREG	General Multiple Regression	29
DAMRCA	Dahlgren Multiple Regression Comprehensive Analysis	35
WEPORU	Uncorrelated Weighted Polynomial Regression	41
WEPORC	Correlated Weighted Polynomial Regression	45
MROP	Multiple Regression Using Orthogonal Polynomials	49
CANON	Canonical Analysis of Second Order Response Functions	57
DURBWAT	Durbin-Watson Test for Independence of Residuals	61
NEARNEB	Near Neighbor Estimation of Experimental Error	63
GOODNESS OF FIT ANALYSIS		67
UNORGOF	Univariate Normal Goodness of Fit	69
BNORGOF	Bivariate Normal Goodness of Fit	75
EXPGOF	Exponential Goodness of Fit	81
WBLGOF	Weibull Goodness of Fit	83
PERGOF	Pearson System Goodness of Fit	87
UNKSGOF	Univariate Normal Kolmogorov-Smirnov Test of Fit	93
RANDOM	Test of Fit for Uniform Random Number Generators	99
POWER EVALUATION		103
DISCRETE POWER EVALUATION		107
BIN1POW	Power of the Test on a Binomial Proportion	109
BIN2POW	Power of the Test on the Difference of Two Binomial Proportions	113
POI1POW	Power of the Test on the Poisson Parameter	121

CONTINUOUS POWER EVALUATION		125
NOR1POW	Power of the One-Sample Normal Test on the Mean	127
NOR2PWE	Power of the Two-Sample Normal Test on Means with Equal Sample Sizes	131
NOR2PWU	Power of the Two-Sample Normal Test on Means with Unequal Sample Sizes	135
T1POW	Power of the One-Sample t Test on the Mean	141
T2POW	Power of the Two-Sample (Pooled) t Test on Means	147
CHIVPOW	Power of the Chi-square Test on the Variance	153
FVARPOW	Power of the F Test for the Equality of Variances	157
FEMPOW	Power of the Test for One-Way Fixed Effects Analysis of Variance	163
REMPOW	Power of the Test for One-Way Random Effects Analysis of Variance	167
PROBABILITY EVALUATION		171
BINVARP	Binomial Probability Distribution with Unequal Single Trial Probabilities	173
NEGBIN	Negative Binomial Probability Distribution	177
CONFIDENCE LIMIT EVALUATION		179
BINCL	Confidence Limits for the Binomial Parameter p	181
CEPCL	Confidence Limits for the CEP (Circular Probable Error)	185
SEPCL	Confidence Limits for the SEP (Spherical Probable Error)	191
MISCELLANEOUS STATISTICAL ANALYSIS		197
LD50EST	Estimation of LD50 (Lethal Dose 50th Percentile)	199
FFAC2K	Analysis of the 2**k Fractional Factorial Experiment	203
SUBROUTINES		211
RANDOM NUMBER GENERATION		213
DISCRETE RANDOM NUMBER GENERATORS		217
RANARB	Arbitrary (User Specified) Discrete Distribution	219
RANBER	Bernoulli Distribution	221
RANBIN	Binomial Distribution	223
RANGEO	Geometric Distribution	225
RANHYP	Hypergeometric Distribution	227
RANNBI	Negative Binomial Distribution	229

RANPOI	Poisson Distribution	231
RANUWO	Discrete Uniform Distribution (Without Replacement)	233
RANUWR	Discrete Uniform Distribution (With Replacement)	235
CONTINUOUS RANDOM NUMBER GENERATORS		237
RANBET	Beta Distribution	239
RANCSQ	Chi-square Distribution	241
RANEXP	Exponential Distribution	243
RANFDI	F Distribution	245
RANGAM	Gamma Distribution	247
RANLGS	Logistic Distribution	249
RANLOG	Lognormal Distribution	251
RANNOR	Normal Distribution	255
RANNVE	Multivariate Normal Distribution	257
RANPDI	Pearson Distributions	261
RANTDI	Student's t Distribution	265
RANUNI	Continuous Uniform Distribution (On a Line)	267
RANCIR	Continuous Uniform Distribution (Within a Circle)	269
RANWEI	Three-parameter Weibull Distribution	271
RANMK1	1st Order Markov Process	273
GLOSSARY		275
DISTRIBUTION		277

INTRODUCTION

OVERVIEW OF STATLIB

STATLIB is a direct access file on the CDC General Purpose Computers at NSWC under UN=LIBRARY. It contains over 50 programs and subroutines for statistical data analysis and random number generation. The programs are initiated interactively via a menu driver (STATMNU) which queries the user for the name of the program he desires to run. User name, charge code, input file and printing instructions are requested and then a batch job is submitted for him accordingly. The results are returned to the user's directory in a file named STATOUT (or a user specified filename) which is created for him.

The subroutines are initiated within the user's main program. Their call lines and arguments are described in this document, but since they are not menu driven, they do not appear in the menu of programs.

ORIGIN OF STATLIB

Statistical analysis has been used at NSWC since the earliest days of ordnance testing. As a result of increased requirements for and increased complexity of statistical analysis, the Mathematical Statistics Branch (currently the Mathematical Statistics Staff, E406) was formed in K Department in 1963. Much of the computation and analysis at that time was conducted on mechanical desk top machines and on programs written on an as needed basis. Reliable commercial software as we know it today simply did not exist. Hence, statistical software became a by-product of the branch almost from its very beginning. At the onset, very little actual programming was done within the branch. The usual approach was to formulate the requirements and obtain programming support from the Computer Programming Division. This process continued well into the 70's when programming was conducted within the branch. The upshot of all this is that many different programmers were involved in the emerging software. With the exception of those programs which are documented in NSWC Reports, the original programmers are usually anonymous. As commercial statistical software became more available, more reliable, and more friendly, the need for new in-house statistical programs was considerably reduced. Currently, only a small effort within the Mathematical Statistics Staff is devoted to software development.

ESTABLISHMENT OF STATLIB

During the phase-out period of the CDC 6700 in late 1984, it was evident that the myriad of statistical programs developed over the years could not and should not be con-

verted to the new general purpose computer (the current CDC 995). Some were designed to compute descriptive statistics and had become obsolete with respect to new commercial software. Some were simply large and lethargic by today's standards and not worthy of the effort involved in conversion and checkout. This left a collection of about 30 programs deemed worthy of retention. In addition, it left a collection of programs that would offer better utilization in the form of subroutines. With the obsolescence of card files, a computer library was considered the most efficient means of storage with rapid access. While Mathematical Statistics Staff members would probably be the prime users of such a library, its establishment would make the software accessible to all NSWC personnel. With respect to the programs, this establishment involved reviewing each program, updating where appropriate, designing a comprehensive test case, and correcting any discovered errors. With respect to the subroutines, it involved a reconfiguration of old programs into subroutines and programming of newly formulated subroutines. (Some of the reconfigured programs were random number generators, and new ones were written to form a complete set of such generators for the common probability distributions.) Both the new and the old were extensively tested for correctness. While error free software is a rare commodity, it is believed that the programs and subroutines which form STATLIB contain very few errors. This is because of the long history and recent testing of the old software and the extreme care taken in developing the new.

The user should bear in mind that this library is not a complete library in the sense of a commercial package. For example, it will not do basic descriptive statistics nor can it be used to categorize or sort data. STATLIB should be regarded as a specialized set of programs and subroutines which form an excellent complement to the commercial statistical packages available at NSWC. Programs in STATLIB either do what the commercial packages do not, do it better, or simply allow the user to have more control over what is done.

COMMERCIAL STATISTICAL PACKAGES AT NSWC

The commercial statistical package which is currently available on the primary mainframes at NSWC is SPSSX. It is an enhanced and extended version of its predecessor, SPSS, which was available at the Center on the CDC 6700 from 1982 through 1985. This package is a comprehensive tool for managing, analyzing, and displaying information. It can take data from almost any kind of file and extract meaningful information using a wide variety of statistical procedures. The authors of this report have found it to be an excellent data management tool. In addition, it contains some very creditable statistical programs, e.g., descriptive statistics, multiple linear regression, and one way analysis of variance are quite good. On the other hand, their treatment of nonlinear regression and non-orthogonal analysis of variance is limited. In brief, SPSSX is an excellent data management tool with

provisions for basic statistical analysis, but for many specialized procedures, one must turn to other commercial packages and STATLIB. Unfortunately, other statistical packages are not compatible with the CDC hardware at NSWC for reasons to be discussed below.

There are three major comprehensive mainframe statistical packages, and they are all geared for IBM and other non-CDC mainframes. In addition to SPSSX, these packages are SAS and BMDP. SAS is an excellent package for statistical analysis which is available at most universities. However, it is written in machine language for non-CDC mainframes and, hence, not available to NSWC. BMDP leases a CDC version of their package, but it has been found to be extremely time consuming to install on NSWC hardware. Therefore, of the major commercial statistical packages available for mainframes, NSWC is essentially limited to SPSSX. Even with SPSSX, conversions of their enhancements for CDC hardware are last on the list. For example, NSWC waited over a year longer for the release of SPSSX than did the users with IBM mainframes.

Versions of all three of the above named packages have recently become available for desk top personal computers. They are referred to as SPSSPC+, BMDPC, and SASPC and in general are less comprehensive than their mainframe counterparts. The basic version of SPSSPC+ and most of the more useful BMDPC programs have been purchased by the Mathematical Statistics Staff. Of course, these are copy protected and subject to copywrite laws which means that they are available at only a single work station.

In addition to the three major statistical packages cited for mainframe and PC use, there are scores of less comprehensive and less widely known statistical software currently available and scores becoming available for personal computers. Some of these are very good and inexpensive and have been purchased by the Mathematical Statistics Staff.

USING STATLIB

LIBRARY ORGANIZATION

STATLIB is comprised of a collection of (1) 34 programs for statistical data analysis and probability calculations and (2) a set of 24 subroutines for random number generation. The programs are subdivided into seven categories; namely, regression analysis, goodness of fit analysis, discrete power evaluation, continuous power evaluation, probability evaluation, confidence limit evaluation, and miscellaneous analysis. The subroutines are grouped into two categories -- discrete random number generators and continuous random number generators.

HOW TO CALL IT

In order to access STATLIB the user must first attach and library STATLIB as follows:

```
ATTACH,STATLIB/UN=LIBRARY.
LIBRARY,STATLIB.
```

If the user wishes to execute a STATLIB program, no other system libraries need be attached. If the user wants access to the STATLIB subroutines, then the system library MATHLIB must also be attached and libraried under UN=LIBRARY.

INFORMATION NEEDED TO RUN IT

The programs in STATLIB are initiated interactively via a menu driver. After attaching STATLIB the command

```
STATMNU
```

will display a menu consisting of the seven program categories. General details on how to proceed at this stage are available to the user through a HELP function key. If no help instructions are required interactively, the user will select the category of programs he wishes to access. A second menu will then appear listing all programs under that category. A general description of any particular program in this category can be obtained at this juncture by using the HELP function key followed by the number of the program for which a description is desired. If no basic program descriptions are needed interactively, the user selects the number of the specific program he wishes to execute.

If the program selected is the first program to be executed in the current interactive session, the user will see the following appear on the screen:

```
                    MAKJCL


                  USER NAME:
                 CHARGE CODE:


          Specify values and press NEXT when ready
```

Once this information has been provided, a second screen will appear:

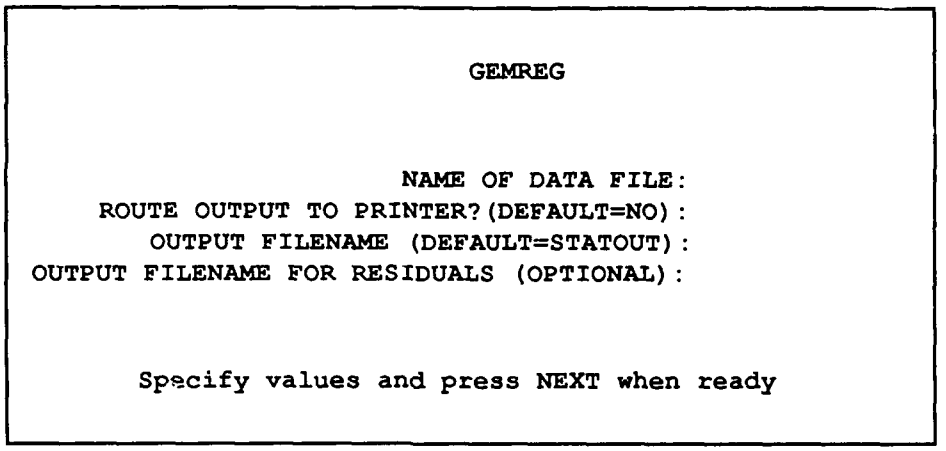
```
                STATLIB PROGRAM NAME


                       NAME OF DATA FILE:
  ROUTE OUTPUT TO PRINTER? (DEFAULT=NO) :
     OUTPUT FILENAME (DEFAULT=STATOUT) :


          Specify values and press NEXT when ready
```

The data file requested is the name of the input file which contains the user's data and input instructions in accordance with the selected program's input guide. On the CDC 995 any name consisting of from 1 to 7 alphanumeric characters which begins with a letter is permissible. The printing option gives the user the flexibility to route the output immediately or to defer printing until the output can be viewed interactively (i.e., input errors may be present or the output may not be as desired). The user is also given the option of specifying the output filename. If no filename is specified, the name STATOUT will be used. Once this program information has been furnished a batch job is submitted and the results returned to the user's permanent directory under the chosen filename. If additional STATLIB programs are executed within the same interactive session, only the second screen above will appear.

Some STATLIB programs have the capability to write useful information onto a data file during execution. For example, the STATLIB regression program GEMREG gives the user the option of creating a data file of model residuals from the regression fit. For these STATLIB programs the second screen will require an additional user input. For example, this screen will appear as follows for program GEMREG:



This last filename request gives the user the flexibility to store the model residuals from the regression fit in the filename of his choice. The input guide for GEMREG specifies the contents and the format of this file of residuals.

A caution regarding the output filename is in order. If the user does not specify an output filename the default name STATOUT will be used. This file will reside in the user's permanent directory. Unless it is purged, it will remain there and be replaced by a new version of STATOUT each time a STATLIB program is executed. It is not recommended that the user purge STATOUT each time a STATLIB program is run. However, care should be exercised when running the same STATLIB program multiple times to ensure that the version of STATOUT being viewed is the version which corresponds to the user's current run. If, for example, the user submits a run whose input file contains errors without realizing it, then, depending on the severity of the error, he may be viewing the most recent copy of STATOUT placed in his permanent directory when he last ran that program successfully.

STATLIB does not provide a menu listing of its subroutines. These subroutines must be called from the user's main program after attaching both STATLIB and MATHLIB.

EXAMPLES

PROGRAM

An example of the execution of one of the programs in STATLIB is provided here as an aid to the user. The program chosen to illustrate STATLIB usage is GEMREG, the first program in the STATLIB category of regression.

The first step in executing any program in STATLIB is the preparation of an appropriate data file. The data file contains the user's data and input instructions in accordance with the selected program's input guide. Program input guides and other related information are given in Section IV, DESCRIPTIONS AND INPUT GUIDES. Prior to execution of the program the data file must be stored in the user's permanent directory.

Next the user must follow the instructions given in Section II, USING STATLIB, which require applying the ATTACH and LIBRARY commands for STATLIB. Then after typing STATMNU the user is requested to choose a program and provide answers to several questions. One of these questions requests the name of the user's data file. After the data file name (and other required information) is given, the program is executed. The output is returned to the user's permanent directory under the file name STATOUT (default) or a user specified file name. The output may also be routed directly to the printer.

Data for the GEMREG example was taken from Example 10.4 on page 362 in ***Probability and Statistics for Engineers and Scientists*** by R. E. Walpole and R. H. Myers, Third Edition, Macmillan Publishing Company. The data consists of 18 observations of two regressor variables, X_1 and X_2 , and a response variable, y. The particular values of X_1 and X_2 represent nine unique design points and each design point was replicated twice yielding 18 data points. A data file containing the 18 data points and input instructions according to the GEMREG input guide was prepared and stored as a permanent file under file name DATGEMX.

The contents of DATGEMX are given on page 11. The reader may wish to refer to the GEMREG input guide as he views that page. The input has been prepared assuming a second order regression model. Predicted values of the response variable were requested. Also, 95% confidence limits for the mean response and a single future response were requested at four synthetic points. A single rerun was indicated with the three second order terms being deleted from the model.

The GEMREG output for the input data file DATGEMX is given on pages 12 - 18. This is the output that would be returned to STATOUT, assuming the default option was chosen. There was no request made to write model residuals onto a second file. The system day file for the GEMREG run is on pages 18 and 19. This file is available to the user under a file name assigned by the system.

SUBROUTINE

Instructions on the use of subroutines in STATLIB are also given in Section II, USING STATLIB. As indicated there the ATTACH and LIBRARY commands must be applied for STATLIB and MATHLIB for access to STATLIB subroutines.

In order to execute a subroutine from STATLIB the user must construct a main or calling program. Argument values are passed between the main program and subroutine through the subroutine call line. Call lines and other descriptive information about each subroutine are given in Section IV, DESCRIPTIONS AND INPUT GUIDES. The user is responsible for printing all argument values of the subroutine, should he wish to see them.

CONTENTS OF DATGEMX DATA FILE

```
GEMREG EXAMPLE
   18     3     3     4     1     0     P      (3F10.2)
 P        1 2   2 2
CP        1 2
CL          4     1                                          .05
          75.          15.
          75.          25.
         125.          15.
         125.          25.
 R          4     5     6
          75.          15.       14.05
          75.          15.       14.93
          75.          20.       16.56
          75.          20.       15.85
          75.          25.       22.41
          75.          25.       21.66
         100.          15.       10.55
         100.          15.        9.48
         100.          20.       13.63
         100.          20.       11.75
         100.          25.       18.55
         100.          25.       17.98
         125.          15.        7.55
         125.          15.        6.59
         125.          20.        9.23
         125.          20.        8.78
         125.          25.       15.93
         125.          25.       16.44
```

```
*********************************************************
***PROGRAM GEMREG : GENERAL MULTIPLE REGRESSION     ***
*********************************************************
***NAVAL SURFACE WARFARE CENTER; DAHLGREN, VIRGINIA***
*********************************************************
***COMPLETION DATE OF THIS VERSION: JULY, 1988      ***
*********************************************************
```

GEMREG EXAMPLE

NUMBER OF OBSERVATIONS - 18

INPUT VARIABLES - X(1) THRU X(3)

DEPENDENT VARIABLE - X(3)

GENERATED VARIABLES -

X(4) = X(1) ** 2
X(5) = X(2) ** 2
X(6) = X(1) * X(2)

DATA MATRIX - (NOTE - POSITION OF ITH ORIGINAL OR GENERATED VARIABLE IS DENOTED BY X(I), X(0) EQUALS 1 FOR ALL OBSERVATIONS)

	X(1) (1)	X(2) (2)	X(4) (3)	X(5) (4)	X(6) (5)	X(3) (6)
(1)	.750000000E+02	.150000000E+02	.562500000E+04	.225000000E+03	.112500000E+04	.140500000E+02
(2)	.750000000E+02	.150000000E+02	.562500000E+04	.225000000E+03	.112500000E+04	.149300000E+02
(3)	.750000000E+02	.200000000E+02	.562500000E+04	.400000000E+03	.150000000E+04	.165600000E+02
(4)	.750000000E+02	.200000000E+02	.562500000E+04	.400000000E+03	.150000000E+04	.158500000E+02
(5)	.750000000E+02	.250000000E+02	.562500000E+04	.625000000E+03	.187500000E+04	.224100000E+02
(6)	.750000000E+02	.250000000E+02	.562500000E+04	.625000000E+03	.187500000E+04	.216600000E+02
(7)	.100000000E+03	.150000000E+02	.100000000E+05	.225000000E+03	.150000000E+04	.105500000E+02
(8)	.100000000E+03	.150000000E+02	.100000000E+05	.225000000E+03	.150000000E+04	.948000000E+01
(9)	.100000000E+03	.200000000E+02	.100000000E+05	.400000000E+03	.200000000E+04	.136300000E+02
(10)	.100000000E+03	.200000000E+02	.100000000E+05	.400000000E+03	.200000000E+04	.117500000E+02
(11)	.100000000E+03	.250000000E+02	.100000000E+05	.625000000E+03	.250000000E+04	.185500000E+02
(12)	.100000000E+03	.250000000E+02	.100000000E+05	.625000000E+03	.250000000E+04	.179800000E+02
(13)	.125000000E+03	.150000000E+02	.156250000E+05	.225000000E+03	.187500000E+04	.755000000E+01
(14)	.125000000E+03	.150000000E+02	.156250000E+05	.225000000E+03	.187500000E+04	.659000000E+01
(15)	.125000000E+03	.200000000E+02	.156250000E+05	.400000000E+03	.250000000E+04	.923000000E+01
(16)	.125000000E+03	.200000000E+02	.156250000E+05	.400000000E+03	.250000000E+04	.878000000E+01
(17)	.125000000E+03	.250000000E+02	.156250000E+05	.625000000E+03	.312500000E+04	.159300000E+02
(18)	.125000000E+03	.250000000E+02	.156250000E+05	.625000000E+03	.312500000E+04	.164400000E+02

SUMMATION MATRIX -

	(0)	(1)	(2)	(3)	(4)	(5)	(6)
(0)	.180000000E+02	.180000000E+04	.360000000E+03	.187500000E+06	.750000000E+04	.360000000E+05	.251920000E+03
(1)	.180000000E+04	.187500000E+06	.360000000E+05	.202500000E+08	.750000000E+06	.375000000E+07	.241685000E+05
(2)	.360000000E+03	.360000000E+05	.750000000E+04	.375000000E+07	.162000000E+06	.750000000E+06	.528750000E+04
(3)	.187500000E+06	.202500000E+08	.375000000E+07	.225468750E+10	.781250000E+08	.405000000E+09	.242073750E+07
(4)	.750000000E+04	.750000000E+06	.162000000E+06	.781250000E+08	.360750000E+07	.162000000E+08	.115135000E+06
(5)	.360000000E+05	.375000000E+07	.750000000E+06	.405000000E+09	.162000000E+08	.781250000E+08	.508672500E+06
(6)	.251920000E+03	.241685000E+05	.528750000E+04	.242073750E+07	.115135000E+06	.508672500E+06	.389627280E+04

CORRELATION MATRIX -

	(1)	(2)	(3)
(1)	.1000E+01	0.	-.6140E+00
(2)	0.	.1000E+01	.7472E+00
(3)	-.6140E+00	.7472E+00	.1000E+01

INVERSE MATRIX AND SOLUTION VECTOR -

	(0)	(1)	(2)	(3)	(4)	(5)	(6)
(0)	.152277778E+03	-.156000000E+01	-.780000000E+01	.613333333E-02	.153333333E+00	.160000000E-01	.564411111E+02
(1)	-.156000000E+01	.289333333E-01	.160000000E-01	-.128000000E-03	-.116683988E-13	-.160000000E-03	-.361933333E+00
(2)	-.780000000E+01	.160000000E-01	.723333333E+00	.201754219E-16	-.160000000E-01	-.800000000E-03	-.275300000E+01
(3)	.613333333E-02	-.128000000E-03	-.222397656E-14	.640000000E-06	.527826704E-16	.108657466E-17	.813333333E-03
(4)	.153333333E+00	-.609910964E-16	-.160000000E-01	-.109656667E-18	.400000000E-03	.314442251E-17	.817333333E-01
(5)	.160000000E-01	-.160000000E-03	-.800000000E-03	-.157969163E-18	.551597154E-16	.800000000E-05	.314000000E-02

IDENTITY MATRIX FOR ACCURACY CHECK ON INVERSE -

	(0)	(1)	(2)	(3)	(4)	(5)
(0)	.1000E+01	.1319E-12	.2363E-12	-.3293E-16	-.7345E-14	-.1329E-14
(1)	-.8238E-09	.1000E+01	.3072E-10	-.2390E-14	-.7995E-12	-.1375E-12
(2)	-.1494E-09	.2626E-11	.1000E+01	-.6609E-15	-.1283E-12	-.2717E-13
(3)	-.6114E-07	.1208E-08	.4046E-09	.1000E+01	-.7040E-11	-.1306E-10
(4)	-.3139E-08	.5432E-10	.8936E-10	-.1409E-13	.1000E+01	-.5693E-12
(5)	-.1447E-07	.2515E-09	.3286E-09	-.6779E-13	-.7235E-11	.1000E+01

REGRESSION EQUATION -

Y = .5644111111E+02 + -.3619333333E+00 X(1) + -.2753000000E+01 X(2) + .8133333333E-03 X(4) + .8173333333E-01 X(5) + .3140000000E-02 X(6)

VALUE OF DETERMINANT = .1977539062E+23

ANOVA

SOURCE	SS	DF	MS	F RATIO
REGRESSION	.36547657E+03	5	.73095314E+02	.1742E+03
RESIDUAL	.50358722E+01	12	.41965602E+00	
LACK OF FIT	.92117222E+00	3	.30705741E+00	.6716E+00
PURE ERROR	.41147000E+01	9	.45718889E+00	
TOTAL	.37051244E+03	17		

CORRELATION COEFFICIENT = .99318093E+00

SQUARE ROOT OF RESIDUAL VARIANCE = .64780863E+00

PREDICTED VALUES

OBSERVATION	Y	YHAT	RESIDUAL
(1)	.140500000E+02	.144986111E+02	-.448611111E+00
(2)	.149300000E+02	.144986111E+02	.431388889E+00
(3)	.165600000E+02	.162144444E+02	.345555556E+00
(4)	.158500000E+02	.162144444E+02	-.364444444E+00
(5)	.224100000E+02	.220169444E+02	.393055556E+00
(6)	.216600000E+02	.220169444E+02	-.356944444E+00
(7)	.105500000E+02	.101861111E+02	.363888889E+00
(8)	.948000000E+01	.101861111E+02	-.706111111E+00
(9)	.136300000E+02	.122944444E+02	.133555556E+01
(10)	.117500000E+02	.122944444E+02	-.544444444E+00
(11)	.185500000E+02	.184894444E+02	.605555556E-01
(12)	.179800000E+02	.184894444E+02	-.509444444E+00
(13)	.755000000E+01	.689027778E+01	.659722222E+00
(14)	.659000000E+01	.689027778E+01	-.300277778E+00
(15)	.923000000E+01	.939111111E+01	-.161111111E+00
(16)	.878000000E+01	.939111111E+01	-.611111111E+00
(17)	.159300000E+02	.159786111E+02	-.486111111E-01
(18)	.164400000E+02	.159786111E+02	.461388889E+00

MINIMUM RESIDUAL = -.706111111E+00
MAXIMUM RESIDUAL = .133555556E+01

SS(RESIDUAL) CHECK = .50358722E+01

SYNTHETIC POINTS

(1)	.750000E+02	.150000E+02	.562500E+04	.225000E+03	.112500E+04
(2)	.750000E+02	.250000E+02	.562500E+04	.625000E+03	.187500E+04
(3)	.125000E+03	.150000E+02	.156250E+05	.225000E+03	.187500E+04
(4)	.125000E+03	.250000E+02	.156250E+05	.625000E+03	.312500E+04

CONFIDENCE LIMITS FOR A GIVEN SET OF SYNTHETIC POINTS
USING PURE ERROR MEAN SQUARE

ALPHA = .0500
T(.0250, 9) = 2.2622

SYNTHETIC POINT	PREDICTED VALUES Y(HAT)	S.E.(YHAT)	CONFIDENCE LIMITS TRUE MEAN VALUE OF Y LOWER	UPPER	1 FUTURE OBS. MEAN OF LOWER	UPPER
(1)	.144986111E+02	.4291E+00	.1353E+02	.1547E+02	.1269E+02	.1631E+02
(2)	.220169444E+02	.4291E+00	.2105E+02	.2299E+02	.2021E+02	.2383E+02
(3)	.689027778E+01	.4291E+00	.5920E+01	.7861E+01	.5079E+01	.8702E+01
(4)	.159786111E+02	.4291E+00	.1501E+02	.1695E+02	.1417E+02	.1779E+02

HAND SELECTED RERUN

VARIABLES DELETED -
X(4)
X(5)
X(6)

DATA MATRIX - (NOTE - POSITION OF ITH ORIGINAL OR GENERATED VARIABLE IS DENOTED BY X(I), X(O) EQUALS 1 FOR ALL OBSERVATIONS)

	X(1) (1)	X(2) (2)	X(3) (3)
(1)	.750000000E+02	.150000000E+02	.140500000E+02
(2)	.750000000E+02	.150000000E+02	.149300000E+02
(3)	.750000000E+02	.200000000E+02	.165600000E+02
(4)	.750000000E+02	.200000000E+02	.158500000E+02
(5)	.750000000E+02	.250000000E+02	.224100000E+02
(6)	.750000000E+02	.250000000E+02	.216600000E+02
(7)	.100000000E+03	.150000000E+02	.105500000E+02
(8)	.100000000E+03	.150000000E+02	.948000000E+01
(9)	.100000000E+03	.200000000E+02	.136300000E+02
(10)	.100000000E+03	.200000000E+02	.117500000E+02
(11)	.100000000E+03	.250000000E+02	.185500000E+02
(12)	.100000000E+03	.250000000E+02	.179800000E+02
(13)	.125000000E+03	.150000000E+02	.755000000E+01
(14)	.125000000E+03	.150000000E+02	.659000000E+01
(15)	.125000000E+03	.200000000E+02	.923000000E+01
(16)	.125000000E+03	.200000000E+02	.878000000E+01
(17)	.125000000E+03	.250000000E+02	.159300000E+02
(18)	.125000000E+03	.250000000E+02	.164400000E+02

SUMMATION MATRIX -

	(0)	(1)	(2)	(3)
(0)	.180000000E+02	.180000000E+04	.360000000E+03	.251920000E+03
(1)	.180000000E+04	.187500000E+06	.360000000E+05	.241685000E+05
(2)	.360000000E+03	.360000000E+05	.750000000E+04	.528750000E+04
(3)	.251920000E+03	.241685000E+05	.528750000E+04	.389627280E+04

CORRELATION MATRIX -

	(1)	(2)	(3)
(1)	.1000E+01	0.	-.6140E+00
(2)	0.	.1000E+01	.7472E+00
(3)	-.6140E+00	.7472E+00	.1000E+01

INVERSE MATRIX AND SOLUTION VECTOR -

	(0)	(1)	(2)	(3)
(0)	.272222222E+01	-.133333333E-01	-.666666667E-01	.110355556E+02
(1)	-.133333333E-01	.133333333E-03	-.347869881E-17	-.136466667E+00
(2)	-.666666667E-01	.256914860E-17	.333333333E-02	.830333333E+00

IDENTITY MATRIX FOR ACCURACY CHECK ON INVERSE -

	(0)	(1)	(2)
(0)	.1000E+01	.7375E-15	.6728E-14
(1)	-.7494E-11	.1000E+01	.6467E-12
(2)	-.7994E-12	.1552E-13	.1000E+01

REGRESSION EQUATION -

Y = .1103555556E+02 + -.1364666667E+00 X(1) + .8303333333E+00 X(2)

VALUE OF DETERMINANT = .4050000000E+08

ANOVA

SOURCE	SS	DF	MS	F RATIO
REGRESSION	.34650967E+03	2	.17325483E+03	.1083E+03
RESIDUAL	.24002778E+02	15	.16001852E+01	
LACK OF FIT	.19888078E+02	6	.33146796E+01	.7250E+01
PURE ERROR	.41147000E+01	9	.45718889E+00	
TOTAL	.37051244E+03	17		

CORRELATION COEFFICIENT = .96706636E+00

SQUARE ROOT OF RESIDUAL VARIANCE = .12649843E+01

PREDICTED VALUES

OBSERVATION	Y	YHAT	RESIDUAL
(1)	.140500000E+02	.132555556E+02	.794444444E+00
(2)	.149300000E+02	.132555556E+02	.167444444E+01
(3)	.165600000E+02	.174072222E+02	-.847222222E+00
(4)	.158500000E+02	.174072222E+02	-.155722222E+01
(5)	.224100000E+02	.215588889E+02	.851111111E+00
(6)	.216600000E+02	.215588889E+02	.101111111E+00
(7)	.105500000E+02	.984388889E+01	.706111111E+00
(8)	.948000000E+01	.984388889E+01	-.363888889E+00
(9)	.136300000E+02	.139955556E+02	-.365555556E+00
(10)	.117500000E+02	.139955556E+02	-.224555556E+01
(11)	.185500000E+02	.181472222E+02	.402777778E+00
(12)	.179800000E+02	.181472222E+02	-.167222222E+00
(13)	.755000000E+01	.643222222E+01	.111777778E+01
(14)	.659000000E+01	.643222222E+01	.157777778E+00
(15)	.923000000E+01	.105838889E+02	-.135388889E+01
(16)	.878000000E+01	.105838889E+02	-.180388889E+01
(17)	.159300000E+02	.147355556E+02	.119444444E+01
(18)	.164400000E+02	.147355556E+02	.170444444E+01

MINIMUM RESIDUAL = -.224555556E+01
MAXIMUM RESIDUAL = .170444444E+01

SS(RESIDUAL) CHECK = .24002778E+02

SYNTHETIC POINTS
(1) .750000E+02 .150000E+02
(2) .750000E+02 .250000E+02
(3) .125000E+03 .150000E+02
(4) .125000E+03 .250000E+02

CONFIDENCE LIMITS FOR A GIVEN SET OF SYNTHETIC POINTS
USING PURE ERROR MEAN SQUARE

ALPHA = .0500
T(.0250, 9) = 2.2622

SYNTHETIC POINT	PREDICTED VALUES Y(HAT)	S.E.(YHAT)	CONFIDENCE LIMITS TRUE MEAN VALUE OF Y LOWER	UPPER	MEAN OF 1 FUTURE OBS. LOWER	UPPER
(1)	.132555556E+02	.3187E+00	.1253E+02	.1398E+02	.1156E+02	.1495E+02
(2)	.215588889E+02	.3187E+00	.2084E+02	.2228E+02	.1987E+02	.2325E+02
(3)	.643222222E+01	.3187E+00	.5711E+01	.7153E+01	.4741E+01	.8123E+01
(4)	.147355556E+02	.3187E+00	.1401E+02	.1546E+02	.1304E+02	.1643E+02

SYSTEM DAYFILE

ABIP (152-01) *NSWC* CYBER 170-875 (MA). NOS 2.5.3 UNCLASS. 89/02/06. 11.06.26.

11.06.17.GGEMMIL.
11.06.17.$USER,GGEMMIL,.
11.06.17.ABSC, B.
11.06.17. YOUR PASSWORD WILL BE EXPIRED 89/12/12.
11.06.17.CHARGE,060013.
11.06.17.$PROLOG,PROC1,,MYUPROC.
11.06.18.$SETFS,PROC1/FS=AD.
11.06.18.PROC1.
11.06.18.$RETURN,PROC1.
11.06.18.$HAFLIFE,30.
11.06.19.MAP,PART.
11.06.19.$SKIP,STOP.
11.06.19.$ENDIF,STOP.
11.06.19.$ATTACH,STARTUP=STARTUP/M=E,UN=LIBRARY,PW=,PN=0.
11.06.19. STARTUP.
11.06.19. NOT INTERACTIVE JOB.
11.06.19. USER DAYFILE PROCESSED.
11.06.19.$RENAME,MYUPROC=ZZZZPR2.
11.06.19.$SETFS,MYUPROC/FS=AD.
11.06.20.$REVERT,EX.$BEGIN,,MYUPROC.
11.06.20.$BEGIN,,MYUPROC.
11.06.20.IFE,0T.EQ.TX0,L01.
11.06.20.ENDIF,L01.
11.06.20.ATTACH,ZZZZZLO=NSWCLIB/UN=LIBRARY.

```
11.06.21.ATTACH,ZZZZZL1=MATHLIB/UN=LIBRARY.
11.06.21.LIBRARY,ZZZZZLO,ZZZZZL1.
11.06.22.REVERT.
11.06.22.GET,TAPE5=DATGEMX.
11.06.22.ATTACH,STATLIB/UN=LIBRARY.
11.06.23.LIBRARY,STATLIB.
11.06.23.REWIND,*,INPUT,OUTPUT.
11.06.23.    6 FILES PROCESSED.
11.06.23.SET,EF=O.  INITIALIZE ERROR FLAG
11.06.23.NOEXIT.
11.06.23.GEMREGB,,STATOUT.
11.06.25.      END  GEMREGB
11.06.25.     131100 MAXIMUM EXECUTION FL.
11.06.25.      0.108 CP SECONDS EXECUTION TIME.
11.06.25.ONEXIT.
11.06.25.IF,EF.NE.O,LABEL1.
11.06.25.ELSE,LABEL1.
11.06.25.COMMENT.  PROGRAM RAN SUCCESSFULLY
11.06.25.REPLACE,STATOUT.
11.06.26.ENDIF,LABEL1.
11.06.26.IF,$$.NE.$$,SAVOUT.
11.06.26.ENDIF,SAVOUT.
11.06.26.UEAD,        O.OO2KUNS.
11.06.26.UEPF,        O.O8OKUNS.
11.06.26.UEMS,        3.108KUNS.
11.06.26.UECP,        O.169SECS.
11.06.26.AESR,        2.842UNTS.
11.06.26.$OUT(*/OP=E)
11.06.26. NO FILES PROCESSED.
11.06.26.$DAYFILE(OUTPUT,JT=D)
```

DESCRIPTIONS

AND

INPUT GUIDES

PROGRAMS

REGRESSION ANALYSIS

Regression analysis is a collection of statistical techniques for investigating and modeling the relationship between variables. For a specific problem, a regression model containing unknown population parameters is postulated. The unknown parameters are estimated from sample data, and statistical tests are performed to ascertain if all the parameters in the postulated model are required, if the model is adequate, or if a relationship between the variables even exists. If the postulated model is linear in the parameters, it is referred to as a linear model. Otherwise, it is referred to as a nonlinear model. For example,

$$y = \beta_0 + \beta_1 X + \beta_2 X^2 + \varepsilon$$

is linear in the parameters and, hence, is a linear model. On the other hand,

$$y = \beta_0 / (1 + \beta_1 e^{-2x}) + \varepsilon$$

is nonlinear in the parameters and, hence, is a nonlinear model. The regression programs in STATLIB are linear regression programs. Programs for nonlinear regression analysis have been purchased commercially. However, it should be noted that a nonlinear model can oftentimes be transformed to a linear one, and linear theory can then be applied to the transformed model. For example,

$$y = \exp(\beta_0 + \beta_1 X + \varepsilon)$$

is nonlinear but intrinsically linear since the transformation

$$\ln y = \beta_0 + \beta_1 X + \varepsilon$$

is linear. Hence, linear regression analysis could be applied to the transformed model.

Linear models in a regression framework come under the heading of the general linear model

$$y = \beta_0 + \beta_1 X_1 + \ldots + \beta_k X_k + \varepsilon.$$

In this model, y is referred to as the dependent (or response) variable, the X_i as independent variables (or regressors), the β_i as regression coefficients, and ε as the random error. The inclusion of ε accounts for the fact that the relationship between the response variable and the regressors is not an exact functional relationship. The parameters β_i are estimated by the method of least squares on the basis of $n > k$ response values y_i for specified values of the independent variables. It is usually convenient to express the general linear model in matrix notation, i.e.,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

where $\mathbf{y}$ is the $n \times 1$ vector of values for the response variable, $\boldsymbol{\beta}$ is the $(k+1) \times 1$ vector of regression coefficients, $\boldsymbol{\varepsilon}$ is the $n \times 1$ vector of random errors, and $\mathbf{X}$ is the $n \times (k+1)$ matrix of values for the regressors. We write

$$\mathbf{X} = \begin{bmatrix} 1 & X_{11} & X_{21} & \dots & X_{k1} \\ 1 & X_{12} & X_{22} & \dots & X_{k2} \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ 1 & X_{1n} & X_{2n} & \dots & X_{kn} \end{bmatrix}$$

where X_{ij} is the value associated with the jth response for the ith regressor. In this notation, the least squares estimates for the β_i are

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

where the 'hat' or circumflex is used to denote the estimate of a parameter.

The problem of estimation in regression is usually straightforward provided the set of influential regressors is known. In most practical problems, this is not the case. Usually, one has a pool of candidate regressors, and the first problem is ascertaining if the pool is sufficient to build an adequate model (lack of fit problem). If the pool is sufficient, one then has the problem of finding the appropriate subset for the model (variable screening problem) via tests of hypotheses. In the final analysis, we have the problem of assessing the worth of the model via confidence limits on the regression coefficients, on the mean response, and on the predicted response. These analyses are based on the fact that the underlying model assumptions have been met. The most critical of these assumptions is that the probability distribution for the vector of errors has covariance equal to $\sigma^2\mathbf{I}$. If they are not met, the analysis becomes more complicated, and the results more difficult to interpret. For further reading on the broad topic of regression, the user is referred to the classical textbooks on the subject, References 1, 2, 3, and 4.

None of the programs in STATLIB are capable of resolving all of the problems in regression, but each one has special features which help resolve some of them. Collectively, they can help the user solve most of his regression problems. Their features and limitations are discussed in the ensuing pages.

REFERENCES

1. Draper, N. R. and Smith, H. (1981), ***Applied Regression Analysis***, Second Edition, John Wiley & Sons, Inc.

2. Graybill, Franklin A. (1961), ***An Introduction to Linear Statistical Models***, Volume I, McGraw-Hill Book Company, Inc.

3. Montgomery, Douglas C. and Peck, Elizabeth A. (1982), ***Introduction to Linear Regression Analysis***, John Wiley & Sons, Inc.

4. Neter, John, Wasserman, William, and Kutner, Michael H. (1985), ***Applied Linear Statistical Models***, Richard D. Irwin, Inc.

GEMREG

PURPOSE

Program GEMREG (General Multiple Regression) is a multiple regression program designed in the late 70's to provide more flexibility and more user options than were available at the time. One of its primary features is its "all regressions" capability which provides for the automatic computation of regression analyses on each regressor alone, each pair, each triplet, etc. In a model with k regressors, this procedure performs 2^k-1 regression analyses. While the program incorporates an "all regressions" testing scheme in an attempt to determine a "best" regression equation, a recent enhancement of the procedure provides a means of determining a "best" regression in the sense of Montgomery and Peck (Reference 1, Chapter 7) and Myers (Reference 2, Chapter 4).

FEATURES

GEMREG features fall into two categories - those standard for all runs and those optional at the discretion of the user. The standard features in GEMREG include the following:

* A correlation matrix showing the simple correlation between all pairs of variables.
* The inverse of $\mathbf{X'X}$ and the solution vector of estimates of the regression coefficients.
* The determinant of $\mathbf{X'X}$ and a check on its inverse.
* An ANOVA (Analysis of Variance) table with a test for regression significance and a test for lack of fit (model adequacy).

Optional features include the following:

* A table of the predicted values and residuals.
* Decomposition of pure error sum of squares.
* Automatic generation of powers and cross product terms.
* Confidence limits on the expected response for specific values of the regressors (referred to as synthetic points).
* Prediction limits on the response for specific values of the regressors (synthetic points).
* "All regressions" testing for the "best" regression equation.

* Computation and tabling of *SSE* (residual sum of squares), *MSE* (residual mean square), R^2 (coefficient of determination), adjusted R^2, and Mallows' C_p statistic for each of the 2^k-1 regressions in the "all regressions" procedure.
* Hand selected reruns to enable the user to test the significance of any subset of regressors.
* Creation of a file in the user's permanent directory containing the observed response y, predicted response $\hat{y}$, and residual $y - \hat{y}$ for all data points in format (E15.9,5X,E15.9,5X,E15.9).

In order to accommodate the "all regressions" feature, the number of regressors (original + generated) is limited to ten. In addition, the number of data points is limited to 1000. The user is referred to Reference 3 for a detailed discussion of the program and to References 1 and 2 for a discussion of regression criteria and the use of "all regressions" measures in determining a "best" regression equation.

REFERENCES

1. Montgomery, Douglas C. and Peck, Elizabeth A. (1982), ***Introduction to Linear Regression Analysis***, John Wiley & Sons, Inc., pp. 244-286.
2. Myers, Raymond H. (1986), ***Classical and Modern Regression with Applications***, Duxbury Press, pp. 101-136.
3. Taub, A. E. and Thomas, M. A. (1981), ***GEMREG-A General Multiple Regression Program***, NSWC TN 81-298, NSWC, Dahlgren, VIRGINIA 22448.

INPUT GUIDE

The user-created input file consists of two records containing standard feature control variables, one or more records containing optional feature control variables, and the data records containing the data points (response values and associated regressor values). The position of the response variable and the input format are controlled by the user on the second control record. Enhancements have been made to the original program since the date of Reference 3. Therefore, in case of discrepancies between the input guide in Reference 3 and the one below the current guide should take precedence.

Record Type	Variable	Description	Columns	Format
1	ID	Problem description.	1-72	9A8

2	NOBS	Number of data points. (NOBS < 1000)	1-5	I5
	NCOL	Number of columns in the data matrix, i.e., the number of variables read in (regressors + response). (NCOL ≤ 11)	6-10	I5
	NDEP	Position of the dependent variable (column number in data matrix).	11-15	I5
	NOP	Number of option records.	16-20	I5
	IPPV	=1, print predicted values. =0, suppress predicted values.	21-25	I5
	IPLOF	=1, print decomposition of pure error sum of squares =0, suppress decomposition printout.	26-30	I5
	MS	=R, use residual mean square for confidence limits and testing. (DEFAULT=R) =P, use pure error mean square for confidence limits and testing.	35	A1
	FORM	Variable format (placed in parentheses) for reading data matrix and synthetic points.	41-80	4A10
3	IOP	Option desired (right justify) =P, power variables will be generated. =CP, cross product variables will be generated. =CL, confidence limits will be computed. =AR, "all regressions" will be generated. =R, hand selected reruns will be executed.	1-2	A2

IOPDIR(I)	If IOP=P, IOPDIR(1) contains the index of the variable and IOPDIR(2) contains the power to which it is to be raised. Repeat as necessary in 5 column sets.	6-10 11-15 16-20 etc.	1X,2I2 1X,2I2 1X,2I2 ditto
	If IOP=CP, IOPDIR(1) and IOPDIR(2) contain the indices of the pairs of variables to be crossed. (More than two variables may be crossed by using indices of generated variables.) Repeat as necessary.	6-10 11-15 16-20 etc.	1X,2I2 1X,2I2 1X,2I2 ditto
	If IOP=CL, IOPDIR(1) contains the number of synthetic points to be read in and IOP-DIR(2) contains the number of future observations in the average for which prediction limits are desired.	6-15	2(1X,I4)
	If IOP=AR, IOPDIR(1)=1 if the printout for each regression is desired. If left blank, new "all regressions" features will not be printed.	10	I1
	If IOP=R, IOPDIR(I) contains the index of the ith variable to be deleted for the rerun. Repeat as necessary.	6-10 11-15 16-20 etc.	1X,I4 1X,I4 1X,I4 ditto
ALPHA	If IOP=CL, 1-ALPHA is the confidence level to be used. If IOP=AR, ALPHA is the significance level to be used for "all regressions" testing.	61-70	F10.0

If the CL option has been specified, record 4 contains the synthetic points (one point per record) according to the format specified by FORM on record 2. This record type is repeated as necessary and must immediately follow record 3 on which the CL option was specified. Omit this record type if the CL option has not been specified.

Record type 5 contains the input data points according to the format specified by FORM on record type 2.

The request for the creation of a file in the user's permanent directory containing y, $\hat{y}$, and $y - \hat{y}$ is not made on any of the input records. This request is made by supplying the permanent file name for this file when prompted during GEMREG setup. The prompt will read OUTPUT FILENAME FOR RESIDUALS (OPTIONAL): _____ . If a name is supplied, the file will be created under that filename.

COMMENTS

If one has control over the input data, it is convenient to place the response variable last. This eliminates problems in exercising the P, CP, and CL options. These options can be exercised if the response is not last, but greater care must be taken in specifying the indices and in formatting the synthetic points.

User selected options (record type 3) should be input in the same order as they are listed in the input guide. The program will run regardless of order. However, if the order shown is not adhered to, there is no guarantee that the input options will be incorporated. For example, options P and CP generate power and cross product terms, respectively. If these options are desired and not input first as listed, then power and cross product terms may not be generated for prior listed options. On the other hand, if they are input first, power and cross product terms will be generated initially and remain in the model through the exercise of all subsequent options unless removed by the R (rerun) option.

DAMRCA

PURPOSE

Program DAMRCA (Dahlgren Multiple Regression Comprehensive Analysis) performs a regression analysis for a general linear model containing up to 50 regressor variables. DAMRCA was written in the early sixties and documented in 1966 in Reference 1. The program contains procedures which allow the analyst to evaluate the statistical significance of a postulated model. A procedure is also available for determining if specified regressors are required in the model. Two ranking procedures are available which provide an ordering of the regressor variables with respect to conditional prediction power for the response variable. By utilizing the procedures within DAMRCA, the analyst can determine a statistically significant model for the response variable and a relative ranking of the regressors for that model.

FEATURES

DAMRCA features fall into two categories - those which are standard for all runs and those which are optional at the discretion of the user. Standard features include the following:

* The inverse and determinant of **X'X** and an accuracy check on the inverse.
* Estimates of the regression coefficients and their standard deviations.
* An analysis of variance table with a test for statistical significance of the regression model.
* Multiple correlation coefficient.
* Predictions and prediction errors (postulated model).
* Histogram and chi-square goodness of fit test for normality of prediction errors (postulated model).

Optional features include the following:

* Automatic generation of powers and cross products of regressors.
* User selected reruns which allow significance testing of any subset of regressors.
* Predictions and prediction errors for reruns.
* Histogram and chi-square goodness of fit test for normality of prediction errors for reruns.

* Predictions and prediction standard deviations for a future response and for the expected response at specified values of the regressors (for construction of confidence limits).
* Forward selection procedure for regressors (IVOR).
* Backward elimination procedure for regressors (BIVOR).

For a detailed discussion of the program, see Reference 1. The forward selection and backward elimination procedures along with many other topics related to regression analysis are discussed in Reference 2.

REFERENCES

1. Abt, K., Gemmill, G., Herring, T. and Shade, R. (1966), ***DA-MRCA: A Fortran IV Program for Multiple Linear Regression***, NWL Report No. 2035, NSWC, Dahlgren, VIRGINIA 22448.
2. Montgomery, D. C. and Peck, E. A. (1982), ***Introduction to Linear Regression Analysis***, John Wiley & Sons, Inc.

INPUT GUIDE

A concise input guide for the input file is given below. A more detailed version is provided in Reference 1. Some changes have been made to the original program since the printing of Reference 1. In the case of discrepancies between the two versions, the current input guide given below should be followed. Record types 1, 2, 8 and 9 are mandatory and comprise the most basic run that a user could submit. The other records listed below afford the user with optional output as indicated.

Record Type	Variable	Description	Columns	Format
1	PGLB	Problem description.	1-72	9A8
2	IR	Number of original regressors.	1-2	I2
		($IR \leq 50$)		
	IS	Number of generated regressors (powers and cross products of input regressors).	3-4	I2
		($IR + IS \leq 50$)		

NR	Number of reruns.	5-7	I3
MVP	Number of synthetic points (points which are not contained in the input data) for which predictions and prediction standard deviations will be computed.	8-10	I3
NDR	Number of input points for which predictions and prediction standard deviations will be computed.	11-13	I3
NPE	=0, predictions, prediction errors and normality test of prediction errors will be calculated for the postulated model only. =1, calculations designated above will be done for reruns and IVOR/BIVOR runs.	15	I1
NDPO	=0, data coordinates will be printed in 9F13.6. Prediction and prediction errors will be printed in 2F15.6. =1, data coordinates will be printed in 7E17.8. Prediction and prediction errors will be printed in 2F15.6. =2, data coordinates will not be printed but prediction and prediction errors will be printed in 2F15.6.	16	I1
IVORGO	=0, IVOR and BIVOR will not be used. =1, IVOR will be used. =2, BIVOR will be used. =3, both IVOR and BIVOR will be used.	18	I1
NFD	Maximum number of variables to be read from each input record for data point coordinates.	19-20	I2
IBID	=0, identity matrix will be computed and checked for accuracy for all BIVOR runs. =1, identity matrix and accuracy computations will terminate in BIVOR after accuracy criterion has been met.	21	I1
TOLI1	Accuracy criterion for identity matrix printout. 0.0001 is suggested value.	23-31	E9.5

TOLI2	Accuracy criterion for identity matrix acceptance. 0.001 is suggested value.	32-40	E9.5
FORM	Format for reading coordinates of data points. No parentheses are needed and specification should ignore columns 1-2 of the input record. The format applies to a single record only. If more than one record is required for a data point, subsequent records should use the same format.	41-80	5A8

Record type 3 is used for generation of regressors from powers and cross products of the original (input) regressors. If no regressors are to be generated this record is omitted. A maximum of 10 regressors can be used to form a product term.

3	IN(1,1)	Subscript of the regressor to be used as the first factor in the first product term.	1-2	I2
	IN(1,2)	Subscript of the regressor to be used as the second factor in the first product term.	3-4	I2
	.....	.....	....	...
	IN(1,10)	Subscript of the regressor to be used as the tenth factor in the first product term.	19-20	I2

The descriptions of the second, third and fourth product terms are entered in columns 21-40, 41-60 and 61-80, respectively, in the same format. If more than four terms are required, additional records are added as needed.

Record type 4 is used to designate IVOR instructions. If IVOR is not used, this record must be omitted.

4	IQ	Number of regressors to be ranked by IVOR.	1-2	I2
	MI	Number of groups into which the regressors are to be divided for ordering within groups. The subdivision into groups is based strictly on the order of input and generation of regressors.	3-5	I3
		(MI $\leq$ 25)		
	NJ(1)	Number of regressors in the first group.	6-8	I3
	NJ(2)	Number of regressors in the second group.	9-11	I3
	.....	.....	....	...

	NJ(25)	Number of regressors in the twenty-fifth group.	78-80	I3

Record type 5 is used to designate BIVOR instructions. If BIVOR is not used, this record must be omitted.

5	MB	The number of groups into which the regressors are to be divided for ordering within groups. As with IVOR, grouping is based on the order of input and generation of the regressors.	1-2	I2
		(MB ≤ 25)		
	LOT(1)	Number of regressors in the first group (last group to be ranked).	3-5	I3
	LOT(2)	Number of regressors in the second group (next to last group to be ranked).	6-8	I3
	....	.....	....	...
	LOT(25)	Number of regressors in the twenty-fifth group (first group to be ranked).	75-77	I3

Record type 6 is used to designate input points for which predictions and prediction standard deviations will be computed. Entries refer to the points according to their order of input and must be in numericaliy ascending order. If no input points are to be selected this record must be omitted.

6	IKEEPR (1)	Number corresponding to the input order of the first selected input point.	1-4	I4
	IKEEPR (2)	Number corresponding to the input order of the second selected input point.	5-8	I4
	.....	.....	....	...
	IKEEPR (20)	Number corresponding to the input order of the twentieth selected input point.	77-80	I4

Additional records are needed if more than 20 points are selected.

Record type 7 is used to input synthetic points for which predictions and prediction standard deviations will be computed. If no synthetic design points are to be used, this record must be omitted. The format specification for reading the coordinates of the data points is also

used to read the synthetic points. Therefore, a record for a synthetic design point should be identical to a record for a data point except that the field for the response variable should be blank.

Record type 8 is used to input the coordinates of the data points. The coordinates are entered according to the format given on record 2 in columns 41-80. This format applies starting in column 2 of the data point input records. The coordinate of the response variable must be entered as the first coordinate for each data point. Maximum number of data points is 9500.

9	MI	=99,	indicating termination of input for data point coordinates.	1-2	I2

Record type 10 is used to designate rerun models. Each rerun must be indicated by a separate record. If no reruns are desired, this record must be omitted.

10	LOT(1)	=0,	include constant in rerun.	1	I1
		=1,	exclude constant from rerun.		
	LOT(2)	=0,	include first regressor in model.	2	I1
		=1,	exclude first regressor from model.		
	.....		.....	....	...
	LOT(51)	=0,	include fiftieth regressor in model.	51	I1
		=1,	exclude fiftieth regressor from model.		

WEPORU

PURPOSE

Program WEPORU (Uncorrelated Weighted Polynomial Regression) is designed to fit a weighted curvilinear model with one independent variable to a set of observations. Program WEPORU handles the case in which the error terms (differences between the observed and predicted values of the dependent variable) have different variances but are uncorrelated. If the error terms are correlated program WEPORC should be used. Program WEPORU complements two other regression programs in STATLIB, namely, GEMREG and DAMRCA. These programs are based on the general linear model and assume that the error terms have the same variance for all observations and are uncorrelated.

FEATURES

WEPORU features can be classified as either standard for all runs or optional based on user preference. Standard features in WEPORU include the following:

* Fitted equation showing the estimated regression coefficients.
* Two ANOVA (Analysis of Variance) tables: one table with a test for regression significance showing the contribution made by each term in the model to the overall regression sum of squares, and the other table displaying a test for lack of fit (model adequacy).
* A table displaying the raw input data, the estimated (predicted) values for the dependent variable, and the residuals.

Optional features include the following:

* Confidence limits on the expected response for specific values of the regressor.
* Prediction limits on the response for specific values of the regressor.

These values of the regressor may be either its original levels (input points) or up to 100 synthetic points (points which are not contained in the input data).

The user specifies the degree of the polynomial fit desired, not to exceed the minimum of the number of observations minus two, or 10. The number of observations processed by WEPORU ranges from 3 to 750. The user must also specify an array of "weights", one weight associated with each distinct level of the independent variable. For each level the weight is usually chosen to be the reciprocal of the variance of the response variable at that

level. If these variances are unknown they must be estimated from current and/or past data. The user is referred to Reference 3 for a detailed discussion of WEPORU (called WEPOR in Reference 3) and to References 1 and 2 for background information on weighted regression.

REFERENCES

1. Draper, N. R. and Smith, H. (1981), ***Applied Regression Analysis***, Second Edition, John Wiley & Sons, Inc., pp. 108 - 116.
2. Myers, R. H. (1986), ***Classical and Modern Regression with Applications***, Duxbury Press, pp. 168 - 177.
3. Shields, P. A. and Thomas, M. A. (1982), WEPOR: A Weighted Polynomial Regression Program, NSWC TR 82-49, NSWC, Dahlgren, VIRGINIA 22448.

INPUT GUIDE

The user must create a data file containing the following record types. Record type 1 specifies a title for problem description. Record type 2 specifies the number of observations, the desired degree of the polynomial model, and the value of an option variable dictating whether or not confidence/prediction limits are desired. The third record type is included only if confidence/prediction limits have been requested. This record specifies the number of synthetic points at which these limits are to be computed, a number between 0 and 1 representing one minus the confidence level associated with these limits, and the number of future observations on which the prediction limits are to be based. Record type 4 contains the format by which the data, the independent and dependent variable values, are to be read. Record type 5 contains the format under which the "weights" are to be read. Record type 6 specifies the values of the weights themselves. Record type 7 contains the input data in pairs - first the independent variable value, then the dependent variable value. This record is repeated as often as necessary. Record type 8 is included only if synthetic points are to be specified. This record contains the values of these synthetic points.

Some changes have been made to the original program since the date of Reference 3. For this reason the input guide specified below should take precedence over the one given in Reference 3.

Record Type	Variable	Description	Columns	Format
1	ITITLE	Problem description.	1-72	9A8

2	NOBS	Number of data points $(3 \leq NOBS \leq 750)$	1-5	I5
	KMAX	Desired degree of polynomial model $(1 \leq KMAX \leq \min(10, NOBS - 2))$	6-10	I5
	COPT	Confidence/prediction limit option. =0, no intervals =1, confidence intervals only =2, confidence and prediction intervals	11-15	I5

Record type 3 is included only if COPT = 1 or 2.

3	NPTS	Number of synthetic points for confidence/prediction limits. Set NPTS = 0 to use input points. $(NPTS \leq 100)$	1-5	I5
	AR	AR = $1-\gamma$ for 100γ percent limits. $(0 < AR < 1.0)$	6-10	F5.2
	M	Number of future observations on which the prediction limits are based. (DEFAULT=1)	10-15	I5
4	FORM1	Format used to read in the (independent, dependent) pairs and the synthetic points	1-80	8A10
5	FORM2	Format used to read in the "weights"	1-80	8A10
6	W	Array of weights		FORM2
7	X	Independent variable level		FORM1
	Y	Dependent variable observation		FORM1

Record 7 is repeated as often as necessary.

Record type 8 is included only if COPT = 1 or 2 and NPTS > 0.

8	XPTS	Synthetic point values	FORM1

WEPORC

PURPOSE

Program WEPORC (Correlated Weighted Polynomial Regression) is designed to fit a weighted curvilinear model with one independent variable to a set of observations. Program WEPORC handles the case in which the error terms (differences between the observed and predicted values of the dependent variable) are correlated. If the error terms are uncorrelated with unequal variances program WEPORU should be used. Program WEPORC complements two other regression programs in STATLIB, namely, GEMREG and DAMRCA. These programs are based on the general linear model and assume that the error terms have the same variance for all observations and are uncorrelated.

FEATURES

WEPORC features can be classified as either standard for all runs or optional based on user preference. Standard features in WEPORC include the following:

* Fitted equation showing the estimated regression coefficients.
* Two ANOVA (Analysis of Variance) tables: one table with a test for regression significance showing the contribution made by each term in the model to the overall regression sum of squares, and the other table displaying a test for lack of fit (model adequacy).
* A table displaying the raw input data, the estimated (predicted) values for the dependent variable, and the residuals.
* The lower triangular portion of the weighting matrix.

Optional features include the following:

* Confidence limits on the expected response for specific values of the regressor.
* Prediction limits on the response for specific values of the regressor.
  These values of the regressor may be either its original levels (input points) or up to 100 synthetic points (points which are not contained in the input data).

The user specifies the degree of the polynomial fit desired, not to exceed the minimum of the number of observations minus two, or 10. The number of observations processed by WEPORC ranges from 3 to 100. The user must provide a variance-covariance matrix for the error terms (i.e, variances along the diagonal and covariances on the off-diagonals).

If the elements of this matrix are unknown, they must be estimated from current and/or past data. A weighting matrix is computed in WEPORC from the input variance-covariance matrix. The user is referred to Reference 3 for a detailed discussion of WEPORC (called WEPOR2 in Reference 3) and to References 1 and 2 for background information on weighted regression.

REFERENCES

1. Draper, N. R. and Smith, H. (1981), ***Applied Regression Analysis***, Second Edition, John Wiley & Sons, Inc., pp. 108 - 116.
2. Myers, R. H. (1986), ***Classical and Modern Regression with Applications***, Duxbury Press, pp. 168 - 177.
3. Shields, P. A. and Thomas, M. A. (1982), ***WEPOR: A Weighted Polynomial Regression Program***, NSWC TR 82-49, NSWC, Dahlgren, VIRGINIA 22448.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies a title for problem description. The second record type specifies the number of observations, the desired degree of the polynomial model, and the value of an option variable dictating whether or not confidence/prediction limits are desired. The third record type is included only if confidence/prediction limits have been requested. This record specifies the number of synthetic points at which these limits are to be computed, a number between 0 and 1 representing one minus the confidence level associated with these limits, and the number of future observations on which the prediction limits are to be based. Record type 4 contains the format by which the data, the independent and dependent variable values, are to be read. Record type 5 contains the format under which the elements of the variance- covariance matrix are to be read. Record type 6 specifies the values of the matrix elements themselves. Record type 7 contains the input data in pairs - first the independent variable value, then the dependent variable value. Record 7 is repeated as often as necessary. Record type 8 is included only if synthetic points are to be specified. This record contains the values of these synthetic points.

Some changes have been made to the original program since the date of Reference 3. For this reason the input guide specified below should take precedence over the one given in Reference 3.

Record Type	Variable	Description	Columns	Format
1	ITITLE	Problem description	1-72	9A8

2	NOBS	Number of data points $(3 \leq NOBS \leq 100)$	1-5	I5
	KMAX	Desired degree of polynomial model $(1 \leq KMAX \leq \min(10, NOBS - 2))$	6-10	I5
	COPT	Confidence/prediction limit option. =0, no intervals =1, confidence intervals only =2, confidence and prediction intervals	11-15	I5

Record type 3 is included only if COPT = 1 or 2.

3	NPTS	Number of synthetic points for confidence/prediction limits. Set NPTS = 0 to use input points. (NPTS ≤ 100)	1-5	I5
	AR	AR = $1-\gamma$ for 100γ percent limits. $(0 < AR < 1.0)$	6-10	F5.2
	M	Number of future observations on which the prediction limits are based. (DEFAULT=1)	10-15	I5
4	FORM1	Format used to read in the (independent, dependent) pairs and the synthetic points	1-80	8A10
5	FORM2	Format used to read in the variance-covariance matrix row by row	1-80	8A10
6	V	Variance-covariance matrix elements		FORM2
7	X	Independent variable level		FORM1
	Y	Dependent variable observation		FORM1

Record 7 is repeated as often as necessary.

Record type 8 is included only if COPT = 1 or 2 and NPTS > 0.

8	XPTS	Synthetic point values	FORM1

MROP

PURPOSE

Program MROP (Multiple Regression Using Orthogonal Polynomials) is a multiple polynomial regression program which uses orthogonal polynomials to estimate the regression coefficients and compute the regression sum of squares. It can accommodate as many as four original regressors (independent variables) and generates the powers and cross product terms internally. The program uses Forsythe's recursive method for generating the orthogonal contrast coefficients (Reference 1) and is applicable to non-equidistant levels of the regressors and proportional frequency (or replication) of the response variable. (Proportional replication is defined in the COMMENTS section.) MROP was written in the mid 60's to eliminate the problem of dealing with the large and/or ill conditioned $\mathbf{X'X}$ matrices associated with multiple polynomial regression models. References 2 and 3 provide the theoretical basis for the application of orthogonal polynomials in regression, and Reference 4 provides a discussion of the use of orthogonal polynomials in least squares surface fitting.

FEATURES

Each MROP run contains a listing of the input values for the regressors and the response variable. Other features are optional and controlled by the user on record 2. These include the following:

* Least squares estimates of the regression coefficients.
* Single degree of freedom mean squares, tests of significance on individual terms, and a test for overall regression significance.
* Printout of the computed orthogonal contrast coefficients.
* An ANOVA (Analysis of Variance) table which treats each regressor as a factor in a factorial analysis.
* Printout of the residuals and a check on the residual sum of squares.
* Backward ranking of the regressors incorporating the admissibility principle and tests to determine a statistically significant model. (The admissibility principle states that if a model contains the pth power of a regressor, it must also contain all powers from 1 to $p-1$. It applies to powers of individual regressors, cross products of regressors, and cross products of powers of regressors.)
* Standard deviations of the mean response and future response for specified values of the regressors (synthetic points) for use in confidence and prediction limit calculations.

* Hand selected reruns to enable the user to generate models which contain subsets of the terms in the main run.

REFERENCES

1. Forsythe, George E. (1957), "Generation and Use of Orthogonal Polynomials for Data-Fitting with a Digital Computer", J. Soc. Indust. Appl. Math., 5, pp. 75-88.
2. Kendall, M. G. and Stuart, A. (1961), ***The Advanced Theory of Statistics***, Vol. 2, Hafner Publishing Company, pp. 356-361.
3. Montgomery, Douglas C. and Peck, Elizabeth A. (1982), ***Introduction to Linear Regression Analysis***, John Wiley and Sons, Inc., pp. 244-286.
4. Thomas, M. A. (1966), ***The Use of Orthogonal Polynomials in Least Squares Surface Fitting over Rectangular Grids***, NWL Technical Memorandum K-1/66, NSWC, Dahlgren, VIRGINIA 22448.

INPUT GUIDE

The user-created input file consists of as many as eleven different records. The first is simply a run identification record, and the second and third contain the number of variables and replicates and the optional feature control variables. Records 4 and 5 contain the formats for reading the input data, 6 contains the regressor values, and 7A and 7B contain the response values. The remaining records contain synthetic points and the rerun specifications if indicated on record 2.

Record Type	Variable	Description	Columns	Format
1	ID	Problem description.	1-72	9A8
2	NIV	Number of original regressors.	1-5	I5
	N1	Highest degree to be generated for the first regressor.	6-10	I5
	N2	Highest degree to be generated for the second regressor.	11-15	I5
	N3	Highest degree to be generated for the third regressor.	16-20	I5
	N4	Highest degree to be generated for the fourth regressor.	21-25	I5
	LN1	Number of levels for the first regressor.	26-30	I5

LN2	Number of levels for the second regressor.	31-35	I5
LN3	Number of levels for the third regressor.	36-40	I5
LN4	Number of levels for the fourth regressor.	41-45	I5
INDREP	Number of equal replications per cell. If unequal, i.e., proportional, use a blank or zero.	46-47	I2
KIND	=1, MSE used to estimate σ.	48-49	I2
	=2, pooled MSE used to estimate σ. (Used in the formation of confidence and prediction limits.)		
TEST	=0 or blank, testing requested.	50-51	I2
	> 0, no testing requested. (Testing associated with model significance.)		
ISD	=0, no printout of orthogonal contrast coefficients.	52-53	I2
	> 0, printout of orthogonal contrast coefficients. (Ordered on increasing degree within increasing level.)		
NSYNPT	Number of synthetic points. (For confidence (prediction) limits on the mean (future) response.)	54-58	I5
ALF	Significance level for testing.	59-70	F12.11
NEQ	Number of hand selected reruns.	71-75	I5
ITAB1	=0 =1 =0 =1	79	I1
IFAC	=0 =0 =1 =1	80	I1
	Table 2 only. Tables 1 and 2. Table 1 only. Table 1 only.		
	(Table 1 = Full factorial analysis ANOVA. Table 2 = Regression analysis ANOVA which results from ranking.)		

Record 3 is not needed for the case of a single regressor. It is required for NIV=2, 3, or 4 for equal or unequal replication.

3	NREP (1,1)	Marginal total number of observations for level 1 of regressor 1.	1-5	I5
	NREP (2,1)	Marginal total number of observations for level 2 of regressor 1.	6-10	I5

	.....	.....	...	...
	NREP (LN1,1)	Marginal total number of observations for level LN1 of regressor 1.	....	I5

Repeat for regressors 2, 3, and 4. Each regressor defines a set, and each set begins on a new record.

4	FMT1	Format for reading the values of the regressors and synthetic points.	1-80	10A8
5	FMT2	Format for reading the values of the response variable.	1-80	10A8
6	X(I,1)	I = 1, 2,..., LN1. the LN1 levels of the first regressor.	....	FMT1
	X(I,2)	I = 1, 2,..., LN2. The LN2 levels of the second regressor.	....	FMT1
	X(I,3)	I = 1, 2,..., LN3. The LN3 levels of the third regressor.	....	FMT1
	X(I,4)	I = 1, 2,..., LN4. The LN4 levels of the fourth regressor.	....	FMT1

Each regressor forms a set, and each set begins on a new record.

Record 7A (I5) contains the number of cell values or replicates to be input on record 7B (FMT2). (This record must be omitted for the case of equal replication, i.e., INDREP > 0.) Record 7B contains all the replicates for that cell. Records 7A and 7B are paired for each cell, and the order of these pairs is implied by the program. It must adhere to the following scheme:

Lay out the data in a multi-way table, and let cell(i,j,k,l) designate the cell containing the response values for the

ith level of regressor 1,
jth level of regressor 2,
kth level of regressor 3,
lth level of regressor 4.

With this arrangement, the order of the input for the response values is based on the rules below.

* Data input begins with cell(i,j,k,l) = (1,1,1,1).

* Index i increases with each cell change until it reaches its maximum value (LN1). The cycle is repeated as often as needed.
* Index j increases with each index i cycle change. The cycle is repeated as often as needed.
* Index k increases with each index j cycle change. The cycle is repeated as often as needed.
* Index l increases with each index k cycle change.

An example will be given in the COMMENTS section.

Record 8 is used only if confidence (prediction) limits on the expected (future) response for specific values of the regressors (synthetic points) are requested on record 2. Record 8 contains the synthetic points in the same format used to read in the regressors.

8	XS(1,(I))	The first synthetic point where (I) is the NIV-tuple specifying the value of each regressor in their natural order, i.e., 1, 2, 3, and 4. (Confusion can be avoided by not over-specifying FMT1.)	....	FMT1
.....		.....	...	...
	XS(NSYNPT,(I))	The last synthetic point.	...	FMT1

Records 9 and 10 are the hand selected rerun records and are used only when reruns are indicated on record 2 (NEQ > 0). Record 9 contains the number of terms to be deleted from the original (full) model. Record 10 contains the description of the terms to be deleted. Records 9 and 10 are paired for each rerun.

9	NTERM	Number of terms to be deleted from the full model.	1-5	I5
10	IN	IN is dimensioned according to the number of regressors in the full model. These integers specify the powers of each regressor or combination of regressors to be deleted. Example: To delete X_2, insert bbb1bbbb where b designates a blank. To delete $X_2^2 \cdot X_3^4$, insert bbb2b4bb.	1-8	4I2

COMMENTS

MROP was written in the pre-"computer friendly" era. However, the data input need not be confusing if one first lays out the data in a multi-way table. This makes it easy to ascertain the correct input order and to obtain the information for Record 3 regarding marginal replication totals. The following example with 3 regressors and 20 responses illustrates the process:

$X_{13} = 60.$

	$X_{11} = 100.$	$X_{21} = 200.$
$X_{12} = 27.$	cell (1,1,1)	cell (2,1,1)
	10.5	8.4
	7.6	8.6
	9.1	6.2
$X_{22} = 42.$	cell (1,2,1)	cell (2,2,1)
	16.9	10.8
$X_{32} = 48.$	cell (1,3,1)	cell (2,3,1)
	6.1	8.5

$X_{23} = 39.$

	$X_{11} = 100.$	$X_{21} = 200.$
$X_{12} = 27.$	cell (1,1,2)	cell (2,1,2)
	15.3	20.3
	17.6	12.6
	14.7	16.3
$X_{22} = 42.$	cell (1,2,2)	cell (2,2,2)
	25.3	22.7
$X_{32} = 48.$	cell (1,3,2)	cell (2,3,2)
	8.9	12.4

Applying the rules for records 7A and 7B, the order of input would be (1,1,1), (2,1,1), (1,2,1), (2,2,1), (1,3,1), (2,3,1), (1,1,2), (2,1,2), (1,2,2), (2,2,2), (1,3,2), (2,3,2). Hence, the first three cycles of records 7A and 7B would be the following: the integer 3 in format I5 for 7A and 10.5, 7.6, 9.1 in format FMT2 for 7B, the integer 3 in format I5 for 7A and 8.4, 8.6, 6.2 in format FMT2 for 7B, and the integer 1 in format I5 for 7A and 16.9 in format FMT2 for 7B.

It will also be instructive to define proportional frequency or replication and apply it to the above example. This term describes a multivariate data set in which the marginal totals of the number of responses determines the number of responses in any particular cell. For the case of 3 regressors,

$$n_{ijk} = [(n_{i..})(n_{.j.})(n_{..k})]/(n_{...})^2$$

where n_{ijk} is the number of responses for the ith level of regressor 1, jth level of regressor 2, and kth level of regressor 3. The $n_{i..}$, $n_{.j.}$, and $n_{..k}$ are the marginal total numbers of responses summed over the dotted subscripts, i.e.,

$$n_{i..} = \sum_j \sum_k n_{ijk}$$

$$n_{.j.} = \sum_i \sum_k n_{ijk}$$

$$n_{..k} = \sum_i \sum_j n_{ijk}$$

Applying this definition to the above example yields

$$n_{1..} = 10 \qquad n_{2..} = 10$$

$$n_{.1.} = 12 \qquad n_{.2.} = 4 \qquad n_{.3.} = 4$$

$$n_{..1} = 10 \qquad n_{..2} = 10.$$

The marginal total numbers of observations must be determined for each case to be run and input on record 3. If the number of replications is not proportional, the output from the program will be erroneous.

CANON

PURPOSE

Program CANON performs a canonical analysis for second order response surface models. Canonical analysis is used as a tool in response surface methodology to transform the estimated response function into canonical form. The canonical form allows the analyst to more easily interpret the estimated relationship between the regressor variables and the response variable. The program can be run using two different forms of input. The input may consist of the coefficients in a second order response function, in which case the usual canonical analysis is performed. An alternative is to input the observed data matrix (design matrix and responses). In this event the response function coefficients, an analysis of variance table and tests of significance for regression and lack of fit are computed in addition to the usual canonical analysis. Reference 1 provides a good discussion of canonical analysis and several numerical examples.

FEATURES

The standard output produced by the program under either input option includes:

* A square matrix B which contains the coefficients from the quadratic form of the estimated response function.
* The characteristic roots of matrix B and the estimated response function in canonical form.
* The location of the stationary point in regressor-space.
* The estimated response at the stationary point and a designation of local maximum, local minimum or saddle point.
* A square matrix M which enables the analyst to relate the original regressors to the new canonical variables.

The additional output produced when the full data matrix is input includes:

* The coefficients of the second order response function.
* An analysis of variance table including significance tests for regression and lack of fit from the second order model.

REFERENCE

1. Myers, R. H. (1971), ***Response Surface Methodology***, Allyn and Bacon, Inc., pp. 67-88.

INPUT GUIDE

The specifications of the user-created input file are given below. Record types 1 and 2 are mandatory. Record types 3 and 4 are necessary if the data matrix input option is chosen. Record type 5 is necessary if the second order response function coefficients are input.

Record Type	Variable	Description	Columns	Format
1	IOP1	=1, input data matrix. =2, input coefficients of second order response function.	1-5	I5
	NO	Number of original regressors, i.e., excluding squared and cross-product terms. (NO ≤ 10)	6-10	I5
2	FORM	Format (in parentheses) for reading data matrix or coefficients from record type 4. If IOP1=1, one data point consisting of NO+1 values will be input per record according to FORM. If IOP1=2, two identifying integers and one coefficient will be input per record. For this case FORM must include an I-format to read two identifying integers.	1-80	8A10
3	P	Number of data points. Omit this record if IOP1=2. (P ≤ 200)	1-5	I5
	If IOP1=1:			
4	X(j), Y j=1,..,NO	Values of regressor variables and response variable according to FORM. Repeat as needed.	1-80	FORM

If IOP1=2:

5	I1,I2,C	Values of two coefficient identifiers and coefficient. I1=i and I2=j represent subscripts of regressors comprising product terms, X_iX_j, in the response function. X_0 is defined equal to 1. (Note: This notation differs from the convention of the other regression programs in the library.) The following example is given for a model with two regressors:	1-80	FORM

I1	I2	Coefficient of
0	0	Constant
1	0	X_1
1	1	X_1^2
2	0	X_2
2	2	X_2^2
1	2	X_1X_2

DURBWAT

PURPOSE

Program DURBWAT performs a Durbin-Watson 2-tailed test of hypothesis on the residuals from a regression fit. The Durbin-Watson statistic is designed to test for the independence of the error terms in the regression model by testing whether the first-order autocorrelation of the residual series is zero. This test can reveal the undesirable presence of a first-order autoregressive autocorrelation structure between residuals. For a detailed discussion of the Durbin-Watson statistic, see References 1 and 2.

FEATURES

Built-in tables limit the number of residuals (N) which can be handled by DURBWAT to the range 15 to 100, inclusive. The regression model from which the residuals came can have up to 5 independent variables (K). Allowable significance levels (ALPHA) for the hypothesis test are 0.10, 0.05, and 0.02.

The values of N, K, and ALPHA are displayed in the output. DURBWAT computes and prints out the following additional quantities:

* The value of the Durbin-Watson test statistic (DWSTAT)
* Values of an upper (DU) and lower (DL) bound
* Values of (4 - DWSTAT) and (4 - DU)

These quantities are used as follows to determine and print out the result of the Durbin-Watson test:

(1) If DWSTAT is less than DL or if (4 - DWSTAT) is less than DL, the test is significant at the chosen ALPHA level.

(2) If DU is less than DWSTAT and DWSTAT is less than (4 - DU), the test is non-significant at the chosen ALPHA level.

(3) Failing (1) or (2), the test is inconclusive at level ALPHA.

REFERENCES

1. Durbin, J. R. and Watson, G. S. (1950, 1951, 1971), "Testing for Serial Correlation in Least Squares Regression", Parts 1 - 3, Biometrika 37 (1950): pp. 409 - 428; 38 (1951): pp. 159 - 178; 58 (1971): pp. 1 - 20.

2. Montgomery, Douglas C. and Peck, Elizabeth A. (1982), ***Introduction to Linear Regression Analysis***, John Wiley & Sons, Inc., pp. 349 - 353.

INPUT GUIDE

The user must first obtain one or more sets of residuals from regression fits in which the response variable has been observed sequentially in time. The user must then create a data file containing the following record types. The first record type specifies the number of cases (ICASES) to be run (each case requiring a separate set of residuals). The second record type specifies the number (K) of independent variables (IV's) upon which the regression was based, the number of data points (N), and the significance level (ALPHA) chosen for the 2-tailed test. Next, the set of time-ordered residuals follows on record type 3, one residual per record. If ICASES is greater than one, record types 2 and 3 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	ICASES	Number of cases	1-5	I5
2	K	Number of IV's	1-5	I5
	N	Number of residuals	6-10	I5
	ALPHA	Significance level	16-20	F5.0
3	Z(1)	1st residual	1-20	F20.0
	Z(2)	2nd residual	1-20	F20.0
	.....	.....	.....	.....
	Z(N)	Nth residual	1-20	F20.0

NEARNEB

PURPOSE

Program NEARNEB (Near Neighbor) performs an interior analysis of the observations in a multiple linear regression problem. Estimates of the coefficients in the regression model are required as input to the program in addition to the data points.

The analysis consists of two procedures. The first procedure identifies remotely located data points in the x-space (or regressor space) which exert considerable influence on the least squares computations. The criterion used to identify the remote points is the value of a statistic designated, WSSDJ, which was proposed in Reference 1. WSSDJ is the weighted sum of the squared standardized distance of point j from the centroid of the x-space. The effect of influential points can then be examined by the analyst after refitting the regression model without these points and observing any differences between the old and new estimates of model coefficients and summary statistics.

The second procedure identifies pairs of data points which are designated as "near neighbors" in the x-space. The criterion used to identify "near neighbors" is the value of a statistic designated, WSSD, which was also proposed in Reference 1. WSSD is the weighted squared standardized distance in x-space between two points. These identified pairs of points are used by the program to compute an estimate of the variance for pure error. This estimate can be compared to the residual mean square from the regression analysis to ascertain if significant lack of fit exists.

The program was obtained from Reference 3. Examples and discussion of the application of the procedures are given in Reference 2.

FEATURES

The program will perform both procedures together or each procedure singly, at the analyst's option.

Output for the procedure to detect remote observations is a list ordered by ascending value of WSSDJ and consisting of:

* Observation sequence number.
* Associated value of WSSDJ.

Output for the procedure to compute an estimate of the standard deviation of pure error consists of two lists. The first list is ordered according to ascending predictions of the dependent variable and consists of:

* Observation sequence number.
* Predicted value of dependent variable and prediction error (residual).
* Delta residual (difference between prediction errors), WSSD value for "near neighbor" candidates and rank order of the 15 smallest WSSD values.

The second list is ordered according to ascending values of WSSD and consists of:

* Cumulative estimates of the standard deviation of pure error.
* WSSD value.
* Observation sequence numbers for the observation pair associated with WSSD value.
* Delta residual.

REFERENCES

1. Daniel, C. and Wood, F. S. (1971), ***Fitting Equations to Data***, John Wiley & Sons, Inc.
2. Montgomery, D. C. and Peck, E. A. (1982), ***Introduction to Linear Regression Analysis***, John Wiley & Sons, Inc., pp. 154-167.
3. Montgomery, D. C., Martin, E. W. and Peck, E. A. (1980), "Interior Analysis of the Observations in Multiple Linear Regression", Quality Technology, Vol. 12, No. 3, pp. 165-173.

INPUT GUIDE

The program was written so that the usual input data file for a multiple regression problem can be easily modified to conform to the input requirements. An input file for NEARNEB can be created by adding two records to the beginning of a file of multiple regression data and one (or more if needed) record at the end. All records are mandatory in the input file.

Record Type	Variable	Description	Columns	Format
1	ITYPE	=1, detect remote observations. =2, estimate error from near neighbors. =3, do both procedures.	1	I1
	NIX	=1, list all possible values of WSSDJ. (4*NOBS - 10 values) =0, list only first 100 values of WSSDJ.	2	I1
	NIND	Number of regressor variables in the data set. (NIND ≤ 50)	3-5	I3
	NAIND	Number of regressors actually in model being analyzed. (Note: this parameter allows user to exclude regressors without rewriting XFORM). (NAIND ≤ NIND)	6-8	I3
	NOBS	Number of observations. (NOBS ≤ 200)	9-11	I3
	NPOSY	The column position of the response variable in the data set.	12-14	I3
2	XFORM	Format (in parentheses) for reading regressor and response variables.	1-80	20A4
3	DAT,YDAT	Values of regressor and response variables corresponding to XFORM and NPOSY. Repeat this record as needed.	1-80	XFORM
4	COEFF	Regression coefficients in the order b_0, b_1, b_2,...,b_{NIND}. Enter 0.0 for coefficients corresponding to regressors in the data set but not presently in the model. Repeat this record as needed.	1-80	8F10.0

GOODNESS OF FIT ANALYSIS

A goodness of fit procedure is a statistical test of hypothesis to determine if a sampled population has a specified probability distribution. In other words and as the name implies, it is a procedure for testing the "goodness of fit" of an observed to a theoretical distribution. A technique commonly used is to construct a histogram from the sampled data and compare it visually with the hypothesized probability distribution, $f(x)$. In constructing the histogram, one chooses subintervals and calculates the ordinate for the ith according to the proportion of observations in subinterval or cell i. This provides for a visual comparison but suffers from the lack of objective criteria to judge whether the data fits the specified distribution. There are two primary procedures which provide this objectivity, and they are Pearson's chi-square goodness of fit test and the Kolmogorov-Smirnov (or K-S) test of fit. Five of the goodness of fit programs in STATLIB are of the former type and, one (UNKSGOF) is of the latter type. These procedures will be discussed in the paragraphs below.

Pearson's chi-square goodness of fit statistic is a measure of discrepancy between the observed cell frequency f and the expected cell frequency e (under the hypothesized distribution) combined over all cells. The test is based on the fact that the statistic

$$C = \sum_{i=1}^{k} (f_i - e_i)^2 / e_i$$

is approximately distributed as a chi-square random variable with $\nu = k - b - 1$ degrees of freedom. In this expression, k is the number of subintervals or cells, and b is the number of parameters in the probability distribution that have to be estimated from the data. To ascertain the risk of falsely rejecting the hypothesized distribution, one compares the statistic C with the percentiles of the chi-square distribution with ν degrees of freedom. The goodness of the approximation depends on the expectations e which must not be too small. Many texts quote 5 as the smallest safe value and recommend the combining of adjoining cells when necessary to achieve this minimum. However, Cochran (References 2 and 3) is much less conservative in his statement, "With unimodal distributions, where expected frequency will be small in the tails, arrange matters so that the minimum expectation in each tail is at least one." Most of the programs in STATLIB provide for user control over the minimum number. The sensitivity or power of the test is dependent on the sample size n, the number of cells k, and how the cells have been formed. In STATLIB, the goodness of fit programs utilizing the chi-square approach either have cells with equidistant boundaries or have cells with equal probability (and hence, with equal expected frequency). In three of the programs, the user can specify his choice. Choosing the equal expectation option usually provides one with a more sensitive test.

The Kolmogorov-Smirnov test of fit is based on a comparison of the sample cumulative distribution function with the cumulative distribution function (cdf) of the hypothesized distribution. The cdf of the hypothesized distribution $f(x)$ is defined by

$$F_0(x) = \int_{-\infty}^{x} f(t)dt$$

and the cdf of the sample is defined by $F_n(x) = i/n$ when x is the ith smallest value in the sample; the K-S test statistic is given by

$$D_n = \max_{\text{all } x} |F_n(x) - F_0(x)| .$$

To ascertain the risk of falsely rejecting the hypothesized distribution, one compares D_n with tabled percentiles of its distribution. The distribution of D_n is independent of $F_0(x)$ provided the hypothesized distribution is completely specified with no unknown parameters. If not, the distribution of D_n depends on $F_0(x)$ and the percentiles must be approximated by Monte Carlo sampling (Reference 6). In general, the K-S test is more sensitive than the chi-square test and can be applied with fewer data points.

The user is referred to References 2, 3, 4, and 5 for amplification on the chi-square test and to References 1, 4, and 6 for amplification on the K-S test.

REFERENCES

1. Bates, Carl B. and Orsulak, Jacqueline R. (1968), ***A Computer Program for the Kolmogorov Goodness of Fit Test for Normality***, NWL Technical Memorandum K-2/68, NSWC, Dahlgren, VIRGINIA 22448.
2. Cochran, W. G. (1952), "The χ^2 Test of Goodness of Fit", Annals of Math. Statistics, 23, p. 315.
3. Cochran, W. G. (1954), "Some Methods of Strengthening the Common χ^2 Tests", Biometrics, 10, p. 417.
4. Bowker, Albert H. and Lieberman, Gerald J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., p. 452.
5. Ledermann, Walter (1984), ***Handbook of Applicable Mathematics***, Volume VI: Statistics, Part A, John Wiley and Sons, p. 358.
6. Lilliefors, Hubert W. (1967), "On the Kolmogorov-Smirnov Test of Normality With Mean and Variance Unknown", J. Am. Stat. Assoc., 62, p. 399.

UNORGOF

PURPOSE

Program UNORGOF (Univariate Normal Goodness of Fit) performs a chi-square goodness of fit test of hypothesis that a random sample of data is from a univariate normal parent population. The density function for the normal distribution is given by

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < \infty$$

where μ and σ^2 are the mean and variance, respectively. The chi-square test statistic is calculated under the assumption that the population parameters, μ and σ^2, are equal to the sample statistics,

$$\bar{x} = \frac{\sum_i x_i}{n} \quad \text{and} \quad s^2 = \frac{\sum_i (x_i - \bar{x})^2}{n-1},$$

respectively, where x_i represents the ith data point and n is the number of data points. Initially, the program attempts to use intervals with equidistant boundaries to perform the test statistic computations. However, this attempt is subject to the restriction that theoretical frequencies for each interval must equal or exceed a user selected value (nominally equal to five). Adjacent intervals are combined to meet this criterion before any goodness of fit calculations are done. A histogram of the sample data is printed which provides the analyst assistance in determining the actual form of the sample distribution, should the hypothesis of normality be rejected. For those situations where the analyst has several samples, the program contains a pooling option. This option allows for testing the individual samples and also for testing a pooled sample constructed from the individual samples. The pooled sample is obtained by first subtracting, for each sample, the sample mean from each data point in the sample. These deviations from the respective sample means are then pooled for all samples, thereby creating a pooled sample which is unaffected by any existing differences between the sample means. This option is especially appropriate for testing normality assumptions for a statistical procedure known as analysis of variance.

Reference 1 contains additional information about the program and its application under its original name of CHITRAN. Reference 2 contains a discussion of the goodness of fit procedure and an example of its application to sample data.

FEATURES

The program allows the analyst to transform the data prior to performing the chi-square goodness of fit test by selecting from eleven transformations.

Standard output for each individual sample includes:

* Mean, standard deviation, range, maximum and minimum of the sample data.
* Histogram of the data with chi-square calculations, observed frequency and theoretical frequency (under the normality hypothesis) for each interval.
* Value of the chi-square test statistic and the degrees of freedom.

Optional output includes similar information to that listed above for the pooled sample and a listing of the pooled sample data.

REFERENCES

1. Gemmill, G. W., Herring, T. L. and Shade R. L. (1967), ***CHITRAN - A 7030 (STRETCH) Computer Program for the Chi-square Test of Normality***, NWL TM K-2/67, NSWC, Dahlgren, VIRGINIA 22448.
2. Wadsworth, G. W. and Bryan, J. G., (1974), ***Applications of Probability and Random Variables***, McGraw-Hill Inc., pp. 388 - 392.

INPUT GUIDE

The specifications of the user-created input file are given below. A more detailed version is provided in Reference 1. However, because of changes to the original program, the input guide in Reference 1 is no longer appropriate. Record types 1, 2, 5, 6 and 7 are mandatory. Record types 3 and 4 are required only if the pooling option is specified. Record type 5 is used to input sample specifications; record type 6 is used for sample identification; and record type 7 is used to input sample data. If more than one sample is to be processed, a set of record types 5, 6 and 7 must be repeated for each sample.

Record Type	Variable	Description	Columns	Format
1	FORM	Format (in parentheses) for reading sample data.	1-80	10A8
2	NOSAM	Number of individual samples. (NOSAM ≤ 500)	1-5	I5

GROUP	=0,	Do chi-square test on individual samples only, i.e., do not pool the samples.	10	I1
	=1,	Do chi-square test on individual samples and on the pooled sample.		
	=2,	Do chi-square test on pooled sample only. Print the observations comprising the pooled sample.		
	=3,	Do chi-square test on pooled sample only. Do not print the observations comprising the pooled sample.		
NOTRAN		Number of transformations to be performed on sample data (including pooled sample).	14-15	I2
IT(1)		An integer designating first transformation for sample data. (Refer to transformation code).	16-20	I5
IT(2)		An integer designating second transformation for sample data.	21-25	I5
.....		.....	....	...
IT(11)		An integer designating eleventh transformation for sample data.	66-70	I5

Transformation Codes

=1,	$X = X$ (no transformation)
=2,	$X = \ln X$
=3,	$X = \ln(\ln X)$
=4,	$X = \ln(1 + X)$
=5,	$X = \ln(1 + \ln(1 + X))$
=6,	$X = \sqrt{X}$
=7,	$X = 1/X$
=8,	$X = 1 + 1/X$
=9,	$X = \arcsin X$
=10,	$X = 2\arcsin\sqrt{X}$
=11,	$X = \arcsin\sqrt{X}$

Record types 3 and 4 refer to the pooling option. If pooling will not be done, they must be omitted from the input file.

3	POOLID	Identification for pooled sample.	1-72	9A8

4	INTOVP	Minimum value for theoretical (expected) frequency for each interval in the chi-square test for the pooled sample. Adjacent intervals are combined to meet this criterion.	6-10	I5
	NOINTP	Number of chi-square tests to be done for the pooled sample, where each test begins with a different number of intervals.	11-15	I5
		(NOINTP ≤ 100)		
	INTERP (1)	Number of intervals for the first chi-square test on the pooled sample.	16-20	I5
		(INTERP ≤ 400)		
	INTERP (2)	Number of intervals for the second chi-square test on the pooled sample.	21-25	I5
	.....	.....	....	...
	INTERP (13)	Number of intervals for the thirteenth chi-square test on the pooled sample.	76-80	I5

If NOINTP > 13, additional records are needed. The entries should be made in accordance with format 16I5.

5	NOBS	Number of data points comprising the first sample.	1-5	I5
		(NOBS ≤ 14,000)		
	INTOV	Minimum value of theoretical (expected) frequency for each interval of the chi-square test for the first sample. Adjacent intervals are combined to meet this criterion.	6-10	I5
	NOINT	Number of chi-square tests to be done for the first sample where each test begins with a different number of intervals.	11-15	I5
		(NOINT ≤ 200)		
	INTER (1)	Number of intervals for the first chi-square test on the sample.	16-20	I5
	INTER (2)	Number of intervals for the second chi-square test on the sample.	21-25	I5
	.....	.....	....	...
	INTER (13)	Number of intervals for the thirteenth chi-square test on the sample.	76-80	I5

If NOINT > 13, additional records are needed. The entries should be made in accordance with format 16I5.

6	IDENT	Identification for the first sample.	1-72	9A8
7	OBSERV	Data for the first sample. Repeat record type 7 as needed to enter all data from first sample.	1-80	FORM

Record types 5, 6, and 7 are repeated in "triples" for each additional sample.

BNORGOF

PURPOSE

Program BNORGOF (Bivariate Normal Goodness of Fit) performs a chi-square goodness of fit test that a random sample of data is from a bivariate normal parent population. The joint density function for the bivariate normal distribution is given by

$$f(x_1,x_2)=\frac{1}{2\pi\sigma_1\sigma_1\sqrt{1-\rho^2}}e^{-Q/2}, \qquad -\infty<x_1<\infty, \quad -\infty<x_2<\infty,$$

where

$$Q=\frac{1}{1-\rho^2}\left[\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2-2\rho\left(\frac{x_1-\mu_1}{\sigma_1}\right)\left(\frac{x_2-\mu_2}{\sigma_2}\right)+\left(\frac{x_2-\mu_2}{\sigma_2}\right)^2\right].$$

The parameters (μ_1,σ_1^2) and (μ_2,σ_2^2) represent the mean and variance for the marginal distributions of x_1 and x_2, respectively. The parameter ρ is the correlation coefficient of x_1 and x_2 and measures the linear dependence between the variables. The chi-square test statistic is calculated under the assumption that the five population parameters, μ_1, σ_1^2, μ_2, σ_2^2 and ρ are equal to the sample statistics,

$$\bar{x}_1=\sum_i x_{1i}/n, \qquad s_1^2=\sum_i(x_{1i}-\bar{x}_1)^2/(n-1),$$

$$\bar{x}_2=\sum_i x_{2i}/n, \qquad s_2^2=\sum_i(x_{2i}-\bar{x}_2)^2/(n-1),$$

$$\text{and} \quad r=\left[\sum_i(x_{1i}-\bar{x}_1)(x_{2i}-\bar{x}_2)\right]/\left[\sum_i(x_{1i}-\bar{x}_1)^2\Sigma_i(x_{2i}-\bar{x}_2)^2\right]^{1/2},$$

respectively. The coordinates (x_{1i}, x_{2i}) represent the ith data point and n is the number of data points. Each data point is classified into one of a set of mutually exclusive intervals. These intervals are areas between coaxial contour ellipses and are determined by arbitrarily chosen probability values under the null hypothesis of bivariate normality. Observed and expected frequencies are then computed for these intervals and the value of the chi-square statistic is obtained in the usual manner.

A detailed explanation of this application of the chi-square goodness of fit criterion is provided in Reference 1. Reference 2 contains a description of the program under its original name of BI-CHI and the results from a numerical example. For a discussion of the bivariate normal distribution the reader is referred to Reference 3.

FEATURES

The program allows the analyst to transform the data prior to performing the chi-square goodness of fit test by selecting from thirteen transformations.

Standard output for each data sample includes:

* A listing of the original sample data and, if appropriate, the transformed sample data.
* The sample minimum, maximum, mean, variance and standard deviation for each variable; the sample correlation coefficient; and the sample regression lines for x_1 on x_2 and x_2 on x_1.
* The upper bound, cumulative probability, observed frequency, theoretical (expected) frequency and chi-square contribution for each interval.
* The value of the chi-square statistic and the degrees of freedom.

Optional output includes:

* The specific categorization of each data point designating within which elliptical interval the point is contained.
* A plot of the sample data which may also display probability contours at the user's option.

REFERENCES

1. Bates, C. B. (1966), ***The Chi-square Test of Goodness of Fit for a Bivariate Normal Distribution***, NWL TM K-77/66, NSWC, Dahlgren, VIRGINIA, 22448.
2. Bates, C. B. and Brown, J. (1967), BI-CHI: ***A Computer Program for the Chi-square Goodness of Fit Test for a Bivariate Normal Distribution***, NWL TM K-72-67, NSWC, Dahlgren, VIRGINIA 22448.
3. Hald, A. (1952), ***Statistical Theory with Engineering Applications***, John Wiley and Sons, Inc., pp. 585 - 621.

INPUT GUIDE

The specifications of the user-created input file are given below. An input guide is also given in Reference 2. However, because of changes to the original program, the input guide in Reference 2 is no longer appropriate. Record types 1, 2, 3, 4, 5 and 9 are mandatory. Record type 1 is used for identification; record type 2 contains the format for reading data points; record type 3 contains the number of goodness of fit tests to be done on the sample data (additional transformations etc.); record type 4 contains program parameters; record type 5 contains probability values for the goodness of fit test intervals; and

record type 9 contains sample data points. Additional goodness of fit tests can be made on the original sample data by repeating record types 4 and 5 plus any desired optional records following the data points on record type 9. Record types 6, 7 and 8 are optional. Record type 6 is used to denote contour ellipses which will be plotted; record type 7 contains a transformation constant for x_1; and record type 8 contains a transformation constant for x_2.

Record Type	Variable	Description	Columns	Format
1	JOB	Problem description	1-72	9A8
2	FMT	Format for reading the data terminator, IEND (see record type 9), and the sample data (include parentheses). IEND precedes the data on each record. If IEND is blank, at least one more data record will follow. On the last data record IEND must equal the number of pairs of (x_1,x_2)-values contained on the record. FMT has the form (I2, ____).	1-80	10A8
3	NRUN	Number of tests to be performed on sample data. If NRUN > 1, records 4, 5 and optional records must be repeated following the data (record 9) for each additional test.	4-5	I2
4	NPRINT	=0, categorization of data is not printed.	5	I1
		=1, categorization is printed.		
	JPLOT	=0, no plotting.	10	I1
		=1, data and contour ellipses are plotted.		
		=2, data only is plotted.		
	ITRAN1	An integer designating the transformation for x_1. (Refer to transformation code.)	14-15	I2
	ITRAN2	An integer designating the transformation for x_2. (Refer to transformation code.)	19-20	I2
	NPAIR	Number of pairs of (x_1,x_2)-values per record.	24-25	I2
	NN	Number of sets of probability values, input on record type 5, defining interval bounds for the chi-square test. See COMMENTS.	30	I1
		(NN ≤ 5)		

	NL	Number of sets of probability values, input on record type 6, defining contour ellipses to be plotted. See COMMENTS.	35	I1

(NL ≤ 5)

Transformation Codes

=1, $X = X$ (no transformation)
=2, $X = \ln X$
=3, $X = \ln(\ln X)$
=4, $X = \ln(A + X)$
=5, $X = \ln(B + \ln(C + X))$
=6, $X = \sqrt{X}$
=7, $X = 1/X$
=8, $X = 1/(D+X)$
=9, $X = \arcsin X$
=10, $X = 2\arcsin\sqrt{X}$
=11, $X = X/E$
=12, $X = \sin X$
=13, $X = \cos X$

Constants A, B, C, D and E are input on record type 7 for x_1 and record type 8 for x_2.

5	PS(1)	The smallest probability value for the 1st set of probability values defining interval bounds for the chi-square tests.	1-5	F5.3
	PDELT(1)	The increment for the 1st set of probability values.	6-10	F5.3
	PE(1)	The largest probability value for the 1st set of probability values.	11-15	F5.3

Additional sets (up to NN ≤ 5, see record type 4) of probability values may be input if additional intervals of unequal probability are to be utilized for the chi-square test. Each additional set is input on this same record in a group of three values, smallest value, increment value and largest value, respectively, according to format 3F5.3. See COMMENTS.

Record type 6 is optional and must be included if plotting will be done, that is, if JPLOT = 0 or 2 on record type 4.

6	PPS(1)	The smallest probability value for the 1st set of probability values defining contours to be plotted.	1-5	F5.3
	PPDELT (1)	The increment for the 1st set of probability values.	6-10	F5.3
	PPE(1)	The largest probability value for the 1st set of probability values.	11-15	F5.3

Additional sets (up to NL ≤ 5, see record type 4) of probability values may be input if variable probability increments between contours are to be utilized for the plots. Each additional set is input on this same record in a group of three values, smallest value, increment value and largest value, respectively, according to format 3F5.3. See COMMENTS.

Record type 7 is optional but must be included if ITRAN1 ≥ 2 on record type 4. If 2 ≤ ITRAN1 ≤ 13 but the transformation selected does not contain one of the constants A, B, C, D or E, a blank record must be included as record type 7.

7	ATRAN1	The A-constant in transformation No. 4 for transforming x_1.	1-14	E14.8
	BTRAN1	The B-constant in transformation No. 5 for transforming x_1.	15-28	E14.8
	CTRAN1	The C-constant in transformation No. 5 for transforming x_1.	29-42	E14.8
	DTRAN1	The D-constant in transformation No. 8 for transforming x_1.	43-56	E14.8
	ETRAN1	The E-constant in transformation No. 11 for transforming x_1.	57-70	E14.8

Record type 8 is optional but must be included if ITRAN2 ≥ 2 on record type 4. If 2 ≤ ITRAN2 ≤ 13 but the transformation selected does not contain one of the constants A, B, C, D or E, a blank record must be included as record type 8.

8	ATRAN2	The A-constant in transformation No. 4 for transforming x_2.	1-14	E14.8
	BTRAN2	The B-constant in transformation No. 5 for transforming x_2.	15-28	E14.8
	CTRAN2	The C-constant in transformation No. 5 for transforming x_2.	29-42	E14.8

	DTRAN2	The D-constant in transformation No. 8 for transforming x_2.	43-56	E14.8
	ETRAN2	The E-constant in transformation No. 11 for transforming x_2.	57-70	E14.8
9	IEND	=blank, this is not the last data record. >0, number of pairs of (x_1,x_2)-values on this, the last record.	1-2	I2
	X1(I) and X2(I)	Input pairs of (x_1,x_2)-values. Data must be input in accordance with format FMT specified on record type 2. Maximum number of pairs is 4000.	3-80	Specified in record type 2

COMMENTS

The program variables NN and NL require additional discussion in order to clarify their usage. NN refers to the number of sets of probability values to be input on record type 5, defining the interval bounds of the chi-square test. NL refers to the number of sets of probability values to be input on record type 6 (optional), defining the contour ellipses to be plotted. Both variables are input according to the same format and are applied in identical fashion; therefore, only NN and its application to the chi-square test is discussed below. However, the comments should be considered to include NL and its application to the plotting of contour ellipses.

To perform a chi-square test with equal probability intervals, the user should set NN=1. Using as an example, a test with .05 probability in each interval, then .05 would be input as the smallest probability bound, .05 would be input as the increment value and .95 would be input as the largest probability bound. This would result in a chi-square test with 20 intervals, each containing a probability of .05. For this case the user should choose an interval probability value which divides one (1) evenly to obtain an integral number of intervals.

If the user wishes to vary the probability within intervals, this can be accomplished by setting NN > 1 (but ≤ 5) and inputting the appropriate probability values in record type 5. This is best illustrated with another example. Consider a case where the user wanted intervals of .05 probability from 0 to .20, intervals of .10 probability from .20 to .80 and intervals of .05 probability from .80 to 1. NN must be set equal to 3 in record type 4. On record type 5, three sets of three probability values each would be input. The first set would be .05, .05 and .20. The second set would be .30, .10 and .80. The third set would be .85, .05 and .95.

EXPGOF

PURPOSE

Program EXPGOF (Exponential Goodness of Fit) performs the chi-square goodness of fit test for the exponential probability density function. The form chosen for the exponential density is

$$f(x) = \frac{1}{\beta} e^{-x/\beta} \quad , \quad x > 0$$

with cumulative distribution function

$$F(x) = \text{Prob}(X \leq x) = 1 - e^{-x/\beta} \quad , \quad x > 0.$$

Parameter β is assumed to be unknown and is estimated with the sample average. This estimate is used in the exponential cumulative distribution function to construct cell boundaries which provide equal cell frequency and hence, equal expectation under the exponential hypothesis.

Further information regarding the exponential distribution is contained in References 1 and 2.

FEATURES

EXPGOF features include the following:

* A listing of the input data.
* A frequency histogram showing the observed frequency in each cell and the chi-square contribution for each cell. Since the expected cell frequencies are equal under the exponential hypothesis, this histogram will appear uniform if the data supports the hypothesis.
* A display histogram based on equidistant cell boundaries. This histogram depicts the shape of the observed frequency distribution.
* The value of the chi-square test statistic and the associated degrees of freedom.
* The capability to perform multiple analyses of the same data set in one computer run.

REFERENCES

1. Johnson, Norman L. and Kotz, Samuel (1970), ***Continuous Univariate Distributions - 1***, Houghton Mifflin Company, pp. 207 - 232.
2. Lindgren, B. W. (1968), ***Statistical Theory***, Second Edition, The Macmillan Company, pp. 33 - 38.

INPUT GUIDE

The user-created input file consists of four record types as described below.

Record Type	Variable	Description	Columns	Format
1	ID	Problem description.	1-72	9A8
2	FORM	Variable format (in parentheses) for reading the sample data.	6-45	4A10
	NTOT	Total number of sample data points.	51-60	I10

Record 3 is used to input the sample data points. The points are entered according to format FORM given on Record 2 in columns 6-45.

4	NTEST	The number of the test.	1-10	I10
	PSUBI	Subinterval probability (equal for each interval).	11-20	F10.5
	NINT2	Number of equidistant intervals to be used in the display histogram.	21-30	I10

Record 4 is repeated for each test to be performed on the data.

WBLGOF

PURPOSE

Program WBLGOF (Weibull Goodness of Fit) performs the chi-square goodness of fit test for the two-parameter Weibull probability density function. The form chosen for the Weibull density is

$$f(x) = \left(\frac{\beta}{\delta}\right)\left(\frac{x}{\delta}\right)^{\beta-1} e^{-(x/\delta)^{\beta}} \quad , \quad x > 0$$

with cumulative distribution function

$$F(x) = \text{Prob}(X \le x) = 1 - e^{-(x/\delta)^{\beta}} \quad , \quad x > 0.$$

Parameters β and δ are both assumed to be unknown and, therefore, must be estimated from the data. The method of maximum likelihood estimation is used for this purpose and provides the following two equations.

$$(n/\beta) - n \ln\delta + \sum_i \ln x_i - \frac{1}{\delta^{\beta}} \sum_i \{x_i^{\beta}(\ln x_i - \ln\delta)\} = 0$$

$$\delta^{\beta} = \frac{1}{n} \sum_i x_i^{\beta}$$

where n is the number of observations in the sample data, x_i is the ith observation, and ln denotes the natural logarithm. These equations are solved numerically for β and δ, and the results designated as $\hat{\beta}$ and $\hat{\delta}$, respectively. The estimates are used in the Weibull cumulative distribution function to construct cell boundaries which provide equal cell frequency and hence, equal expectation under the Weibull hypothesis.

Further information regarding the Weibull distribution, its origin, and applications are contained in References 1, 2, and 3.

FEATURES

WBLGOF features include the following:

* A listing of the input data.
* A frequency histogram showing the observed frequency in each cell and the chi-square contribution for each cell. Since the expected cell frequencies are equal under the Weibull hypothesis, this histogram will appear uniform if the data supports the hypothesis.

* A display histogram based on equidistant cell boundaries. This histogram depicts the shape of the observed frequency distribution.
* Estimates of the Weibull parameters and their variances.
* The value of the chi-square test statistic and the associated degrees of freedom.
* The capability to perform multiple analyses of the same data set in one computer run.

REFERENCES

1. Johnson, Norman L. and Kotz, Samuel (1970), ***Continuous Univariate Distributions - 1***, Houghton Mifflin Company, pp. 250 - 271.
2. Plait, Alan (1962), "The Weibull Distribution - With Tables", Industrial Quality Control, November, 1962.
3. Weibull, Waloddi (1951), "A Statistical Distribution Function of Wide Applicability", Journal of Applied Science, September, 1951.

INPUT GUIDE

The user-created input file consists of four record types as described below.

Record Type	Variable	Description	Columns	Format
1	ID	Problem description.	1-72	9A8
2	FORM	Variable format (in parentheses) for reading the sample data.	6-45	4A10
	NTOT	Total number of sample data points.	51-60	I10

Record 3 is used to input the sample data points. The points are entered according to format FORM given on Record 2 in columns 6-45.

4	NTEST	The number of the test.	1-10	I10
	PSUBI	Subinterval probability (equal for each interval).	11-20	F10.5
	BTOL	Tolerance used in the numerical solution of the maximum likelihood equations.	21-30	F10.5
	NINT2	Number of equidistant intervals to be used in the display histogram.	31-40	I10

STPSIZ	Stepsize (=0.0 to print the display histogram over the entire range of data).	41-50	F10.5

Record 4 is repeated for each test to be performed on the data.

COMMENTS

The input variables (1) BTOL and (2) STPSIZ on record 4 require some elaboration.

(1) The Newton-Raphson procedure is used to numerically solve the likelihood equations. This procedure requires a user supplied tolerance to compare with the difference in the recursive solutions. This tolerance is input as BTOL on record 4. A value of .001 is probably adequate for most goodness of fits. However, if one is using WBLGOF as a means of numerically extracting maximum likelihood estimates for the Weibull distribution, he would want to input a substantially lower tolerance.

(2) The display histogram is controlled by input variables NINT2 and STPSIZ on record 4. If NINT2=10 and STPSIZ=0.0, for example, the data range is divided into 10 cells with equidistant boundaries. In dealing with Weibull data, one oftentimes has several observations which are substantially larger than the remaining body of data. The largest of these values determines the range and, hence, the cell boundary distance. If this value is large enough, it can force the remaining body of data into one or two cell which yields a poor display histogram. If this happens on the first pass, one can control it on subsequent passes by setting STPSIZ to an appropriate value for the distance between cell boundaries. This will truncate data points with extreme values to provide a better display of the bulk of the data.

PERGOF

PURPOSE

Program PERGOF (Pearson System Goodness of Fit) determines which distribution, if any, from the Pearson system of frequency curves best fits a set of data. Any distribution which is determined by its mean (μ) and its second, third, and fourth central moments (μ_2, μ_3, μ_4) is a member of the Pearson family of distributions. μ_2 represents the distribution variance. μ_3 is related to the degree of skewness (lack of symmetry about the mean) of the distribution while μ_4 is related to the amount of kurtosis (peakedness) that the distribution exhibits. This family contains distributions which are bell-shaped, J-shaped, L-shaped, and U-shaped. Within these four general shapes is a continuum of skewed, flattened, and peaked curves. More importantly, almost any set of data can be fit to a Pearson distribution by merely equating the moments μ, μ_2, μ_3, and μ_4. Measures of skewness and kurtosis are given by the following expressions

$$\beta_1 \text{ (skewness)} = \mu_3^2/\mu_2^3$$

$$\beta_2 \text{ (kurtosis)} = \mu_4/\mu_2^2$$

For purposes of reference we point out that the values of skewness and kurtosis for the widely-used normal distribution (symmetric and bell-shaped) are $\beta_1 = 0$ and $\beta_2 = 3$, respectively. The Pearson system provides the analyst with an excellent source of distributions which depart from normality in varying degrees of skewness and kurtosis. Another important quantity, used in the classification of the distribution types in the Pearson system, is the kappa "criterion." This quantity is a function of β_1 and β_2 and is given by the expression

$$\kappa \text{ (kappa)} = \frac{\beta_1(\beta_2+3)^2}{4(2\beta_2-3\beta_1-6)(4\beta_2-3\beta_1)}$$

PERGOF can fit data to nine of the distribution types that are contained in the Pearson system. Three of these (Types I, IV, and VI) are called main types while the remaining six (Types II, III, V, VII, X, and XIII) are referred to as transition types. We note that Type XIII is the well-known normal distribution. The user can choose to force fit the data to any of these nine types or let the program determine the best fitting distribution from among the three main types. If the latter option is selected, the choice is based on the value of the kappa criterion. (If $\kappa < 0$ Type I is the indicated choice; if $0 < \kappa < 1$ Type IV is indicated; and if $\kappa > 1$ Type VI is selected.)

For further details regarding the Pearson system the user should consult References 1 and 2.

FEATURES

The quality of the fit of the input data to one of the Pearson type distributions is assessed by means of the chi-square goodness of fit test of hypothesis. Two options are available to the user for computing the chi-square statistic: (1) equal probability intervals, or (2) equal length intervals. The user must consult a table of the chi-square distribution in order to interpret the value of this statistic. The results of the chi-square test may depend upon which interval option is chosen.

PERGOF output features include the following:

* Input specifications and the range of the input data.
* Estimates of the mean, variance, μ_3, and μ_4.
* Estimates of β_1, β_2, and κ.
* A table of chi-square test results for the chosen interval option. This table displays for each interval the theoretical and observed frequencies, the chi-square contribution, and the right-hand interval boundary in both original and standardized units. The standardized boundary is computed as the original boundary value minus the sample mean.
* The value of the chi-square statistic and a statement identifying the Pearson distribution type to which the fit was made.
* A graph of the chosen Pearson distribution type (asterisks) superimposed on a graph of the input data. Location of the data is indicated graphically by horizontal dashed lines bounded by "+" signs. The functional form of the chosen distribution is displayed below the graph.

PERGOF allows the user to request up to ten fits to the input data, i.e., nine force fits (one to each of the nine available distribution types) and one fit in which the program selects the best main type. In the event that more than one fit is requested, results for all the equal probability interval cases are printed first, followed by results for the equal length interval cases.

PERGOF also has a provision for pooling individual contributions to the chi-square statistic when the equal length interval option is chosen. The user specifies a minimum expected (theoretical) frequency. If the expected frequency in any interval is less than this value, adjacent intervals are combined until this condition no longer exists. Interpretation of the chi-square statistic must then be made with respect to the new number of intervals.

REFERENCES

1. Elderton, W. P. and Johnson, N. L. (1969), ***Systems of Frequency Curves***, Cambridge University Press, pp. 35 - 95.

2. Taub, A. E. (1974), ***DURG - A Documentation of the Dahlgren Universal Random Number Generator***, NWL TN-K-17/74, NSWC, Dahlgren, VIRGINIA 22448.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of observations in the data set to be fit, the number of equal probability and equal length interval cases to be processed for the chi-square statistic, the number of distributional fits to be performed, the minimum expected frequency for pooling, the maximum number of iterations to be used in the numerical integration procedure for determining the expected frequencies, and a flag value indicating whether or not a problem identification is to be supplied. The second record type specifies the format under which the input data is to be read. The third record type is included only if the equal probability interval option is to be exercised. This record specifies up to 28 different probability values for equal probability intervals. The fourth record type is included only if the equal length interval option is to be exercised. This record specifies up to 28 different numbers of intervals for equal length intervals. Record type five is included only if the number of distributional fits to the input data has been specified. (If not, PERGOF will select the best fitting Pearson main type.) This record contains the numbers of the Pearson distribution types to be fitted. Here the value 0 requests the best fitting main type. Up to 10 values may be specified. Record type six is included only if the user has elected to supply a problem identification. This record provides that identification. Record type seven contains the input data according to the format provided by the user.

Input specifics are provided in the input guide below:

Record Type	Variable	Description	Columns	Format
1	NPERCD	Total number of observations in the input data set. (NPERCD ≤ 25000)	1-10	I10

	NPCHI	Number of cases to be processed with the equal probability interval option. The probability values are specified in record type 3. (NPCHI ≤ 28)	11-20	I10
	NKCHI	Number of cases to be processed with the equal length interval option. The values for the numbers of intervals are specified in record type 4. (NKCHI ≤ 28)	21-30	I10
	NDIST	Number of Pearson distribution fits to be made to the input data. (NDIST ≤ 10)	31-40	I10
	POOL	Minimum expected (theoretical) frequency used as the criterion for pooling individual interval contributions to the overall chi-square statistic.	41-50	E10.4
	MAXINT	Maximum number of iterations (DEFAULT = 100) to be used in the numerical integration scheme for determining the expected (theoretical) frequencies. (1 ≤ MAXINT ≤ 100)	51-60	I10
	IDFLAG	Problem identification option. >0, identification will be supplied in record type 6. =0 or blank, no identification will be supplied.	61-70	I10
2	FMT	Format by which the input data is to be read.	1-72	9A8
		Record type 3 is included only if NPCHI > 0.		
3	PCHI(1)	1st probability value to be processed with the equal probability interval option.	1-10	E10.4
	PCHI(2)	2nd probability value to be processed with the equal probability interval option.	11-20	E10.4
	.....	.....	....	...

	PCHI(7)	7th probability value to be processed with the equal probability interval option.	61-70	E10.4

Since NPCHI ≤ 28, up to four record type 3's may be required, each adhering to the above format.

Record type 4 is included only in NKCHI > 0.

4	KCHI(1)	1st value for the number of equally spaced intervals to be processed with the equal length interval option.	1-5	I5
	KCHI(2)	2nd value for the number of equally spaced intervals to be processed with the equal length interval option.	6-10	I5
	.....	.....	....	...
	KCHI(14)	14th value for the number of equally spaced intervals to be processed with the equal length interval option.	66-70	I5

Since NKCHI ≤ 28, up to two record type 3's may be required, each adhering to the above format.

Record type 5 is included only if NDIST > 0.

5	MDIST(1)	1st Pearson distribution fit to the input data. =0, select the Pearson main type which provides the best fit. =1, 2, 3, 4, 5, 6, 7, 10, or 13 force fits to the specified Pearson distribution type (where distribution type Roman numerals have been replaced by their Arabic equivalents).	1-5	I5
	MDIST(2)	2nd Pearson distribution fit to the input data.	6-10	I5
	.....	.....	....	...
	MDIST(10)	10th Pearson distribution fit to the input data.	46-50	I5

Record type 6 is included only if IDFLAG > 0.

6	OBSID	Problem identification.	1-72	9A8

Record type 7 contains the input data according to the format specified by FMT on record type 2.

COMMENTS

Simpson's quadrature formula is the numerical integration technique employed in PERGOF to find required areas associated with the Pearson distributions. Since the Pearson system contains such a wide variety of shapes and Simpson's rule is a generalized method, it follows that the numerical integration scheme will not perform adequately for all possible sets of input data Furthermore, for a given set of data, it is possible that the integration scheme will be adequate for one of the chi-square interval options but not the other. This means that if PERGOF fails to handle the user's input data for one choice of the interval option, the other option should be tried before abandoning PERGOF as an analysis tool. Should the numerical integration scheme in PERGOF fail to handle a given data set an appropriate message is printed for the user.

While PERGOF allows the user to fit a set of data to one of nine Pearson distribution types, subroutine RANPDI is available in STATLIB to permit the user to generate random numbers from one of these types.

UNKSGOF

PURPOSE

Program UNKSGOF (Univariate Normal Kolmogorov-Smirnov Goodness of Fit) performs a Kolmogorov-Smirnov (K-S) test of hypothesis which assesses the agreement of a sample cumulative distribution function with that of the cumulative distribution function (cdf) of the normal distribution. Consider a random variable X with a continuous cdf, $F(x)$. Given a random sample of n observations, the sample cdf, $F_n(x)$, is defined by

$$F_n(x_{(i)}) = i/n$$

where $x_{(i)}$ is the ith smallest value in the sample. The cdf of the normal distribution is given by

$$F(x) = \text{Prob}(X \le x) = \int_{-\infty}^{x} f(t)dt$$

where

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad , \quad -\infty < x < \infty$$

and μ and σ^2 are the mean and variance, respectively of the distribution. These parameters are assumed to be unknown and unspecified in the normality hypothesis, and hence, are estimated from the data by the sample statistics

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad \text{and} \qquad s^2 = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \bar{x})^2 .$$

These estimates are then substituted for μ and σ^2 in $f(x)$ above in order to evaluate $F_0(x)$, the cdf under the normal hypothesis. The K-S test statistic is based on the maximum absolute deviation of $F_n(x_{(i)})$ from $F_0(x_{(i)})$ and is given by

$$D_n = \max_i |F_n(x_{(i)}) - F_0(x_{(i)})| .$$

The percentiles of the distribution of D_n have been approximated by Monte Carlo sampling (Reference 3). A table of critical values at the α level of significance appears in this reference and is included in the "COMMENTS" section of UNKSGOF to enable the user to conduct the required test of hypothesis. The hypothesis that a random sample of n observations comes from an underlying normal distribution is tested by comparing D_n with the

critical value, $D_{n,\alpha}$, from the table. α is the probability of falsely rejecting the hypothesis. If $D_n > D_{n,\alpha}$, reject the hypothesis at the level of significance; otherwise, do not reject the hypothesis.

Unlike the chi-square test, the K-S test requires no grouping of the data and it is applicable to very small samples ($n \geq 4$). Further discussion of the K-S test can be found in References 1, 2, and 3.

FEATURES

UNKSGOF allows the user to transform the sample data prior to performing the K-S test for normality by specifying one of thirteen available transformations.

UNKSGOF output features include the following:

* Input specifications and the original and transformed data.
* Sample statistics: minimum and maximum values, range, sample size, mean, variance, and standard deviation.
* A table of the K-S test computations for each sample value. Included in the table is a listing of the transformed data values in ascending order, their standardized values (i.e., subtract sample mean and divide by sample standard deviation), and the corresponding theoretical and sample cdf values along with their absolute differences.
* The value of the K-S test statistic together with the number of the observation in the ordered sample which yielded this maximum absolute deviation.

UNKSGOF also allows the user to analyze multiple sets of data in a single computer run.

REFERENCES

1. Bates, Carl B. and Orsulak, Jacqueline R. (1968), ***A Computer Program for the Kolmogorov Goodness of Fit Test for Normality***, NWL Technical Memorandum K-2/68, NSWC, Dahlgren, VIRGINIA 22448.
2. Bowker, Albert H. and Lieberman, Gerald J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 454 - 458.
3. Lilliefors, Hubert W. (1967), "On the Kolmogorov-Smirnov Test of Normality With Mean and Variance Unknown", Journal of the American Statistical Association, Vol. 62, pp. 399 - 402.

INPUT GUIDE

Some changes have been made to the original program since the date of Reference 1. For this reason the input guide specified below should take precedence over the one given in Reference 1. The user must create a data file containing the following record types. The first record type specifies the problem description and the second specifies the format under which the input data is to be read. The third record type specifies the number and identity of the transformations to be performed on the input data as well as the number of data values per record. The fourth record type is included only if transformations 4, 5, 8, or 11 are selected by the user. It provides the values of constants required in specifying these transformations. Record type five is used to input the sample data. If more than one set of sample data is to be subjected to the K-S test for normality in the same computer run, record types 1 through 5 must be repeated for each such set.

Input specifics are provided in the input guide below:

Record Type	Variable	Description	Columns	Format
1	JOB	Problem description.	1-72	9A8
2	FMT	Format used to read in the data terminator, ITEST (see record 5), and the sample data (include parentheses). ITEST precedes the data on each record. If ITEST is blank, at least one data value follows on that record. On the last record ITEST must equal the number of observations on that record. FMT has the form (I2,_____).	1-80	8A10
3	NRUN	Number of transformations to be performed on the sample data.	1-2	I2
		$(1 \leq NRUN \leq 13)$		
	ITRAN (1)	=0, do not perform transformation number 1.	11	I1
		=1, perform transformation number 1. (Refer to the list of transformations given below.)		
	ITRAN (2)	=0, do not perform transformation number 2.	12	I2
		=1, perform transformation number 2.		
	.....	.....	....	...

ITRAN (13)	=0,	do not perform transformation number 13.	23	I1
	=1,	perform transformation number 13.		
NBR	Number of data values per record.		26-27	I2
NC	=0,	record type 4 is not included.	30	I1
	=1,	record type 4 is included.		

Transformation Number	Transformation Codes
1	$X = X$ (no transformation)
2	$X = \ln X$
3	$X = \ln(\ln X)$
4	$X = \ln(A + X)$
5	$X = \ln(B + \ln(C + X))$
6	$X = \sqrt{X}$
7	$X = 1/X$
8	$X = 1/(D+X)$
9	$X = \arcsin X$
10	$X = 2\arcsin\sqrt{X}$
11	$X = X/E$
12	$X = \sin X$
13	$X = \cos X$

Record type 4 is included only if trasformations 4, 5, 8, or 11 have been requested by the user.

4	A	Constant in transformation number 4.	1-14	F14.6
	B	Constant in transformation number 5.	15-28	F14.6
	C	Constant in transformation number 5.	29-42	F14.6
	D	Constant in transformation number 8.	43-56	F14.6
	E	Constant in transformation number 11.	57-70	F14.6

Although UNKSGOF does not require that the sample size (n) be input, it must fall in the following range:

$$4 \leq n \leq 2500$$

5	ITEST	=blank, this is not the last record containing sample observations.	1-2	I2
		>0, number of sample observations on this, the last record.		

XX	Array containing sample input data. Data must be input in accordance with format FMT specified in record type 2.	3-80	Specified in record type 2

COMMENTS

In the output of UNKSGOF, results of the K-S test for normality include two columns of absolute values labelled (1) $F_0(j) - F_n(j-1)$ and (2) $F_0(j) - F_n(j)$. The reason that there are two rather than one such columns is that the K-S test is a two-sided test of hypothesis. The rejection region for the test is in two parts, a lower rejection region and an upper rejection region. For each data value a value of D_n is computed for both regions. D_n takes the value given by (1) above for the lower region and (2) for the upper region. The K-S test statistic is the maximum of these two values taken over all the data values. The interested user can consult Reference 1 for complete details.

To enable the user to interpret the value of the K-S test statistic, D_n, computed by UNKSGOF, a table of critical values is required. Such a table is provided below and is taken from Reference 3. Let D_n denote the required critical value, where n is the sample size and α is the level of significance of the hypothesis test. If D_n equals or exceeds the tabled value for $D_{n,\alpha}$, the hypothesis of normality is rejected. Otherwise we fail to reject the hypothesis.

$D_{n,\alpha}$ CRITICAL VALUES FOR THE K-S TEST FOR NORMALITY[1]

α : Level of Significance

Sample size n	0.20	0.15	0.10	0.05	0.01
4	.300	.319	.352	.381	.417
5	.285	.299	.315	.337	.405
6	.265	.277	.294	.319	.364
7	.247	.258	.276	.300	.348
8	.233	.244	.261	.285	.331
9	.223	.233	.249	.271	.311
10	.215	.224	.239	.258	.294
11	.206	.217	.230	.249	.284
12	.199	.212	.223	.242	.275
13	.190	.202	.214	.234	.268
14	.183	.194	.207	.227	.261
15	.177	.187	.201	.220	.257
16	.173	.182	.195	.213	.250
17	.169	.177	.189	.206	.245
18	.166	.173	.184	.200	.239
19	.163	.169	.179	.195	.235
20	.160	.166	.174	.190	.231
25	.142	.147	.158	.173	.200
30	.131	.136	.144	.161	.187
Over 30	$\frac{.736}{\sqrt{n}}$	$\frac{.768}{\sqrt{n}}$	$\frac{.805}{\sqrt{n}}$	$\frac{.886}{\sqrt{n}}$	$\frac{1.031}{\sqrt{n}}$

1 Reprinted by permission from "On the Kolmogorov-Smirnov Test of Normality With Mean and Variance Unknown" by Lilliefors, Hubert W., Journal of the American Statistical Association, Vol. 62 (1967), pp. 399-402.

RANDOM

PURPOSE

Program RANDOM evaluates the usefulness of candidate pseudo-uniform random number generators which have been designed for use on any computer system. The candidate generator must produce random variates which purportedly come from a continuous uniform distribution over the interval 0 to 1. Program RANDOM subjects a sequence of pseudo-uniform numbers from the candidate generator to a collection of statistical "tests of randomness." These tests are designed to expose departures from the assumptions of independence and uniformity for the generated sequence of random variates. The results of these tests can be used to judge the adequacy of the candidate generator.

FEATURES

Program RANDOM performs 11 different statistical tests of hypothesis on a single sequence of 10000 pseudo-uniform random numbers produced by the candidate generator. Each test is conducted at the five percent level of significance and the decision "FAIL TO REJECT" or "REJECT" the appropriate hypothesis appears as part of the output for each test. The tests performed are:

1. Mean and Variance tests
2. Frequency test
3. Kolmogorov-Smirnov (K-S) test
4. Maximum of t test
5. Gap test
6. Poker test
7. Coupon collector's test
8. Permutation test
9. Runs test
10. Serial test for successive pairs
11. Serial correlation test

A frequency table and a graph of the sample cumulative distribution function are displayed in conjunction with the output from the frequency test.

While program RANDOM helps to identify bad generators, we caution that a candidate generator should not necessarily be discarded just because it fails one or two of the tests for randomness since, in our case, each statistical test permits failure for a good generator five percent of the time. If one or more tests fail, it is recommended that program RANDOM be rerun one or more times using different input sequences of size 10000, each generated by a different starting seed value, in order to obtain a better feel for the candidate generator's reliability.

For a detailed discussion of the above tests as well as the interpretation of their results, consult Reference 1.

REFERENCES

1. Crigler, J. R. and Shields, P. A. (1982), ***RANDOM: A Computer Program for Evaluating Pseudo-Uniform Random Number Generators***, NSWC TR 82-93, NSWC, Dahlgren, VIRGINIA 22448.
2. Knuth, D. E. (1981), ***The Art of Computer Programming: Volume 2/Seminumerical Algorithms***, Second Edition, Addison-Wesley, pp. 59 - 73.

INPUT GUIDE

Some changes have been made to the original program since the date of Reference 1. For this reason the input guide specified below should take precedence over the one given in Reference 1.

The user must first generate a sequence of exactly 10000 random variates from the candidate generator. The required data file then consists of one record type repeated 10000 times, each record containing one of the generated random variates. More specifically, the data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	ITEM(1)	1st random variate	1-22	E22.14
	ITEM(2)	2nd random variate	1-22	E22.14
	.....	.....	....	...
	ITEM (10000)	10000th random variate	1-22	E22.14

Any floating point number placed in the first 22 columns of a record will satisfy the above format specification for each variate value. The user is cautioned, however, that if numbers in E format are included in the data file, their exponents must be right-justified to span the field length of 22.

POWER EVALUATION

Statistical hypothesis testing is a decision making process which tells the experimenter whether to accept or reject hypotheses regarding characteristics of the sampling population, i.e., the population from which a sample is extracted. These characteristics are usually stated in terms of the parameters in the probability distribution which governs the sampling population. For this discussion, θ will be used to designate the population parameter of interest. The hypothesis under test is referred to as the null hypothesis and designated H_0. The hypothesis complementary to H_0 is referred to as the alternative hypothesis and denoted by H_1. Using this notation, a generic test of hypothesis can be stated as follows:

$$H_0: \theta \in \omega$$

$$H_1: \theta \in \omega' .$$

In this statement, ω is simply a subset of Ω , the set of admissible values for θ and $\omega' = \Omega - \omega$. For example, if θ represents the population variance σ^2, then Ω is the set of positive real numbers. If ω is taken as the subset of reals in the interval (0,4), then ω' is the set of reals greater than or equal to 4.

To test the hypothesis that H_0 is true, a random sample of size n is taken from the population and a numerical value, referred to as the test statistic, T_0 is computed. A critical region R is then determined such that the decision rule rejects H_0 (and accepts H_1) if $T_0 \in$ R and accepts H_0 otherwise. Ideally, $T_0 \in$ R whenever H_1 is true and $T_0 \notin$ R whenever H_0 is true. However, this kind of test does not exist unless one samples the entire population. In general, our decision rule is subject to two kinds of error:

Type I error: Reject H_0 when H_0 is true
Type II error: Accept H_0 when H_1 is true

The probabilities of committing these errors are referred to as α and β, respectively. That is,

α = Prob(committing a Type I error)
= Prob(rejecting H_0 when H_0 is true)
= Prob($T_0 \in$ R when $\theta \in \omega$)

β = Prob(committing a Type II error)
= Prob(accepting H_0 when H_1 is true)
= Prob($T_0 \notin$ R when $\theta \in \omega'$).

The first of these, α , is controlled by the experimenter through the size of the critical region R. The second error, β , is a function of α, n, and a specific value of θ. Its computation is usually involved even with the aid of high speed computers. The complement of β, 1 - β, is referred to as the power of the test and is simply the Prob(rejecting H_0 when H_1 is true). Hence, low β is synonomous with high power.

Oftentimes, experimenters make strong conclusions from test results only upon rejecting H_0. In this situation, a specific value for power is of little interest. However, if strong conclusions are also to be made upon accepting H_0, then the power of the test is of interest and one must have a means to evaluate it. In fact, for planned experiments, the usual criterion for selecting sample size is to ensure that the power of the test is sufficiently high for a given value or values of θ. While it is usually not practical to solve for n for given values of α, β, and θ, one can evaluate power for a range of values of n and select the sample size on the basis of these results. References 1, 2, 5 and 7 provide additional information on the general theory and application of hypothesis testing and the role of power in sample size selection.

To aid the experimenter in selecting the sample size n for his experiment, power curves have been prepared by various authors for various types of tests. See, for example, references 1, 3, and 8. Power curves show power $(1 - \beta)$ as a function of the parameter under test for specified values of n and α. (In some cases, operating characteristic curves have been prepared which show β instead of $1 - \beta$. These curves convey the same information as power curves and, hence, fall into the category of power curves.) Curves of this type are helpful to the experimenter, but they have some serious limitations. First, families of curves for different values of n are usually placed on the same graph. While this provides information relating test sensitivity to sample size, the many curves placed on a single graph makes it very difficult to find power for values of θ which are close to the boundary of ω and ω'. (In the last example, this would refer to values of σ^2 close to 4.) Second, power curves are available for only the more frequently used values of α, usually .05 and .01. If it were necessary to evaluate power for values of α other than these common values, one would have to resort to extensive computation. And third, power curves are not available from a single source which, in some cases, requires the experimenter to search the literature to find the set of curves appropriate for his experiment.

The power programs in STATLIB have been prepared with the intent of eliminating the power curve problems addressed above. Power is presented in numerical tabular form which eliminates any problem with reading values. In addition, power can be computed for any user selected values of $0 < \alpha < 1$. And lastly, STATLIB provides the experimenter with the capability to compute power for most of the common statistical tests and for some of the "not so common" tests. In most cases, this will eliminate the need to search the literature for the appropriate power curve in sample size determination.

The power programs in STATLIB are divided into two sections, one for continuous probability distributions and one for discrete. In the continuous section, the critical point (beginning of the critical region in the tail of the distribution) for the test is not printed. These values are readily available as percentiles of the common sampling distributions, i.e., the normal, Student's t, chi-square, and the F distributions. Furthermore, each page of tabular power values is associated with many values of n, each of which has a different critical point. Hence, the user is referred to statistical tables (Reference 4, for example) for the critical point appropriate for his test. In the discrete section, the critical points are printed since they are not readily available in statistical tables. These critical values are the

largest (smallest) integers such that the critical regions are at most α. The problem here is that the test statistic T_0 for a discrete test is an integer and hence, the critical point is restricted to an integer. Therefore, one can specify $\alpha = .05$ but may obtain a critical region much smaller than .05. To provide for the user who requires a discrete critical region of size α exactly, the discrete power programs provide a randomized test option (Reference 6). In addition to a critical point, this procedure requires the computation of a value $0 < \lambda < 1$. The decision rule is to reject H_0 if T_0 is greater (less) than the critical point, reject H_0 with probability λ if T_0 is equal to the critical point, and accept H_0 otherwise. If this option is employed, the λ values are printed as well as the critical points, and the power is computed on the basic of a critical region of exact size α.

REFERENCES

1. Bowker, Albert H. and Lieberman, Gerald J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 163 - 265.
2. Brownlee, K. A. (1965), ***Statistical Theory and Methodology In Science and Engineering***, Second Edition, John Wiley and Sons, Inc., pp. 97 - 121.
3. Ferris, Charles L., et al (1946), "Operating Characteristics for the Common Statistical Tests of Significance", Annals of Mathematical Statistics, June, 1946.
4. Hald, A. (1952), ***Statistical Tables and Formulas***, John Wiley and Sons, Inc.
5. Hodges, J. L., Jr. and Lehmann, E. L. (1964), ***Basic Concepts of Probability and Statistics***, Holden-Day, Inc., pp. 323-352.
6. Krutchkoff, Richard G. (1970), ***Probability and Statistical Inference***, Gordon and Breach Science Publishers, pp. 216 - 231.
7. Lehmann, E. L. (1986), ***Testing Statistical Hypotheses***, Second Edition, John Wiley and Sons, Inc.
8. Pearson, E. S. and Hartley, H. O. (1951), "Charts of the Power Functions for Analysis of Variance Tests, Derived from the Non-Central F-Distribution", Biometrika, Vol. 38.

DISCRETE

POWER

EVALUATION

BIN1POW

PURPOSE

Program BIN1POW (Binomial Power, 1 Population) determines the critical (rejection) region and evaluates the power function for tests of hypotheses regarding the proportion of successes p in a binomial population. The tests are based on a random sample of size n from the population assuming that p remains constant from trial to trial and that the trial outcomes (success or failure) are mutually independent. The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are shown below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

T_0 = X − number of successes in n binomial trials.

The probability distribution of T_0 depends on the hypothesized value of the parameter p (p_0 or $p_1 \neq p_0$). For $p = p_i$, this distribution will be denoted by $g(t_0 \mid p_i)$ and has form

$$g(t_0 \mid p_i) = \binom{n}{t_0} p_i^{t_0}(1-p_i)^{n-t_0}, \qquad t_0 = 0, 1, 2, \ldots, n$$

The critical region for the test is based on $g(t_0 \mid p_0)$ and the power function is expressed in terms of $g(t_0 \mid p_1)$.

* *

Hypothesis (1): $H_0: p = p_0$
$H_1: p \neq p_0$.

Critical Region: Reject H_0 if $T_0 < c_1$
Reject H_0 with probability γ_1 if $T_0 = c_1$
Reject H_0 if $T_0 > c_2$
Reject H_0 with probability γ_2 if $T_0 = c_2$

where c_1 is the largest integer such that

$$B_1 = \sum_{t_0=0}^{c_1-1} g(t_0 \mid p_0) \leq \alpha/2$$

c_2 is the smallest integer such that

$$B_2 = \sum_{t_0 = c_2 + 1}^{n} g(t_0 \mid p_0) \quad \leq \quad \alpha/2$$

and

$$\gamma_1 = (\alpha/2 - B_1) / g(c_1 \mid p_0) \qquad \text{(randomized test)}$$
$$\gamma_2 = (\alpha/2 - B_2) / g(c_2 \mid p_0) \qquad \text{(randomized test)}$$
$$\gamma_1 = \gamma_2 = 0 \qquad \text{(non-randomized test)}$$

Power Function:

$$\Pi(p_1, n) = \sum_{t_0 = 0}^{c_1 - 1} g(t_0 \mid p_1) + \gamma_1 g(c_1 \mid p_1)$$

$$+ \sum_{t_0 = c_2 + 1}^{n} g(t_0 \mid p_1) + \gamma_2 g(c_2 \mid p_1) \quad .$$

* *

Hypothesis (2): $H_0: p \leq p_0$
$H_1: p > p_0.$

Critical Region: Reject H_0 if $T_0 > c$
Reject H_0 with probability γ if $T_0 = c$
where c is the smallest integer such that

$$A_1 = \sum_{t_0 = c + 1}^{n} g(t_0 \mid p_0) \quad \leq \quad \alpha$$

and

$$\gamma = (\alpha - A_1) / g(c \mid p_0) \qquad \text{(randomized test)}$$
$$= 0 \qquad \text{(non-randomized test)}$$

Power Function:

$$\Pi(p_1, n) = \sum_{t_0 = c + 1}^{n} g(t_0 \mid p_1) + \gamma g(c \mid p_1) .$$

* *

Hypothesis (3): $H_0: p \geq p_0$
$H_1: p < p_0.$

Critical Region: Reject H_0 if $T_0 < c$
Reject H_0 with probability γ if $T_0 = c$
where c is the smallest integer such that

$$A_2 = \sum_{t_0=0}^{c-1} g(t_0 \mid p_0) \quad \leq \quad \alpha$$

and

$\gamma = (\alpha - A_2) / g(c \mid p_0)$ (randomized test)

$= 0$ (non-randomized test)

Power Function: $\Pi(p_1, n) = \sum_{t_0=0}^{c-1} g(t_0 \mid p_1) + \gamma_1 g(c \mid p_1)$.

FEATURES

BIN1POW features include the following:

* Computation of the critical value c and randomized constant γ for up to 50 values of the sample size n and any one of the 3 hypotheses.
* Computation of power function for each value of n and up to 15 values of $p = p_1$.

REFERENCES

1. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 275 - 276.
2. Krutchkoff, Richard G. (1970), ***Probability and Statistical Inference***, Gordon and Breach Science Publishers, pp. 225 - 226.

INPUT GUIDE

The user-created input file consists of two record types as described below.

Record Type	Variable	Description	Columns	Format
1	NHYP	=1, hypothesis (1) =2, hypothesis (2) =3, hypothesis (3)	1-5	I5
	NTYPE	=1, randomized test =2, non-randomized test	6-10	I5

NSS	Number of N(I) values (n values) to be read in on record type 2. (NSS ≤ 50)	11-15	I5
K	Number of p_1 values requested via $p_1 = p_0 \pm$ i*DELTA, i=0,...,K for NHYP =1 and via $p_1 = p_0$ + i*DELTA, i=0,...,K for NHYP = 2,3. K ≤ 7 for NHYP = 1 K ≤ 14 for NHYP = 2,3.	16-20	I5
DELTA	Increment for generation of p_1 values.	21-30	F10.5
PSUB0	Hypothesized value of p, i.e., p_0.	31-40	F10.5
ALPHA	Significance level α	41-50	F10.5

Record type 2 contains the NSS N(I) values in format 10I8. As many as 5 type 2 records may be required.

BIN2POW

PURPOSE

Program BIN2POW (Binomial Power, 2 Populations) determines the critical (rejection) region and evaluates the power function for tests of hypotheses regarding the difference in the proportions of success in two binomial populations based on random samples of sizes m and n from populations 1 and 2, respectively. For population i (i = 1,2), it is assumed that the proportion of success p remains constant from trial to trial and that the trial outcomes (success or failure) are mutually independent. The table below summarizes the notation regarding the test variables.

Population	Proportion of Success	Number of Successes	Number of Failures	Sample Size
1	p_1	X	$m - X$	m
2	p_2	Y	$n - Y$	n
		$T = X + Y$	$m + n - T$	$m + n$

This table shows that X and Y can realize integral values from 0 to m and 0 to n, respectively, so that the sample space can be designated by an $(m + 1) \times (n + 1)$ array. The method of analysis (Reference 4) forms the critical region on the array by working in terms of the conditional distribution of Y given T for T = 0,1,2,...,m+n. The power of the test is then obtained by summing probabilities over the rejection region for specific values of the p_i under the alternative hypothesis. See the COMMENTS section for an example which shows the rejection region in the two-way sample space. The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are shown below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

Y = number of successes in the sample of size n from Population 2.

The conditional probability distribution of Y given $T = t$ has the form

$$P(Y = y \mid X + Y = t) = \frac{\binom{m}{t-y}\binom{n}{y}}{\binom{m+n}{t}}, \quad \begin{matrix} \max(0, t-m) \le y \le \min(t, n) \\ t = 0, 1, 2, \ldots, m+n \end{matrix}$$

when the null hypothesis H_0: $p_1 = p_2$ is true. The critical region for the test is based on this distribution for each value of T. The power of the test is not dependent on T and is obtained by summing probabilities over the rejection region. These unconditional probabilities are based on the joint probability distribution of X and Y which has the form

$$P_1(X=x, Y=y) = \binom{m}{x}\binom{n}{y} p_1^x (1-p_1)^{m-x} p_2^y (1-p_2)^{n-y} \quad , \quad \begin{matrix} x = 0,1,\ldots,n \\ y = 0,1,\ldots,m \end{matrix}$$

where the subscript 1 on P indicates specific values for p_1 and p_2 under the alternative hypothesis. The power functions below are expressed in terms of $P_1(x, y)$.

* *

Hypothesis (1): H_0: $p_1 = p_2$
H_1: $p_1 \neq p_2$.

Critical Region for $T = t$:

Reject H_0 if $Y < y_L$

Reject H_0 with probability γ_L if $Y = y_L$

Reject H_0 if $Y > y_U$

Reject H_0 with probability γ_U if $Y = y_U$

where y_L is the largest integer such that

$$B_1 = \sum_{y=\min y}^{y_L - 1} P_0(y \mid t) \quad \leq \alpha/2$$

y_U is the smallest integer such that

$$B_2 = \sum_{y=y_U+1}^{\max y} P_0(y \mid t) \quad \leq \alpha/2$$

and

$$\gamma_L = (\alpha/2 - B_1) / P_0(y_L \mid t) \qquad \text{(randomized test)}$$

$$\gamma_U = (\alpha/2 - B_2) / P_0(y_U \mid t) \qquad \text{(randomized test)}$$

$$\gamma_L = \gamma_U = 0 \qquad \text{(non-randomized test)}$$

Power Function:

$$\Pi(p_1, p_2, m, n) = \sum_{t=0}^{m+n} [POW_L(t) + POW_U(t)]$$

where

$$POW_L(t) = \sum_{y=\min y}^{y_L-1} P_1(t-y,y) + \gamma_L P_1(t-y_L,y_L)$$

$$POW_U(t) = \sum_{y=y_U+1}^{\max y} P_1(t-y,y) + \gamma_U P_1(t-y_U,y_U)\,.$$

* *

Hypothesis (2): $H_0: p_1 \le p_2$
$H_1: p_1 > p_2.$

Critical Region for $T = t$: Reject H_0 if $Y < y_L$

Reject H_0 with probability γ_L if $Y = y_L$

where y_L is the largest integer such that

$$A_1 = \sum_{y=\min y}^{y_L-1} P_0(y \mid t) \quad \le \alpha$$

and

$$\gamma_L = (\alpha - A_1) / P_0(y_L \mid t) \qquad \text{(randomized test)}$$

$$\gamma_L = 0 \qquad \text{(non-randomized test)}$$

Power Function:

$$\Pi(p_1,p_2,m,n) = \sum_{t=0}^{m+n} POW_L(t)$$

where

$$POW_L(t) = \sum_{y=\min y}^{y_L-1} P_1(t-y,y) + \gamma_L P_1(t-y_L,y_L)$$

* *

Hypothesis (3): $H_0: p_1 \ge p_2$
$H_1: p_1 < p_2.$

Critical Region for $T = t$: Reject H_0 if $Y > y_U$

Reject H_0 with probability γ_U if $Y = y_U$

where y_U is the smallest integer such that

$$A_2 = \sum_{y=y_U+1}^{\max y} P_0(y \mid t) \quad \le \alpha$$

and

$$\gamma_U = (\alpha - A_2) / P_0(y_U \mid t) \qquad \text{(randomized test)}$$

$$\gamma_U = 0 \qquad \text{(non-randomized test)}$$

Power Function: $\Pi(p_1, p_2, m, n) = \sum_{t=0}^{m+n} POW_U(t)$

where

$$POW_U(t) = \sum_{y=y_U+1}^{\max y} P_1(t-y, y) + \gamma_U P_1(t - y_U, y_U) .$$

FEATURES

BIN2POW features include the following:

* Computation of the critical values and randomized constant for sample sizes $m + n \leq 850$ and any one of the 3 hypotheses.
* Computation of the power function for all pairs of p_1 and p_2 where the limitation on the number of values of each p_i is 25. This will provide as many as 625 evaluations of power in a single run.
* A plot of equal power contours in the (p_1, p_2) domain for as many as 10 values of power
* Computation of conditional power for specified realizations t of T and as many as 32 values of the parameter $\rho = p_2 q_1 / q_2 p_1 \quad (q_i = 1 - p_i)$. See the COMMENTS section for a discussion of this feature.
* A plot of the conditional power as a function of ρ for each value of $T = 0,1,...,m+n$.
* Capability to perform multiple cases in a single run.

REFERENCES

1. Bennett, B. M. and Hsu, P. (1960), "On the Power Function of the Exact Test for the 2 X 2 Contingency Table", Biometrika, Vol. 47.
2. Finney, D. J. (1948), "The Fisher-Yates Test of Significance in 2 X 2 Contingency Tables", Biometrika, Vol. 35.
3. Fisher, R. A. (1967), ***Statistical Methods for Research Workers***, Hafner Publishing Co., Inc., pp. 85 - 99.

4. Lehmann, E. L. (1959), ***Testing Statistical Hypotheses***, John Wiley and Sons, Inc., pp. 140 - 143.

5. Robertson, W. H. (1966), "Programming Fisher's Exact Method of Comparing Two Percentages", Technometrics, Vol. 2.

INPUT GUIDE

The user-created input file consists of as many as 10 record types. The first three are required for all runs. Record types 4 and 5 are optional for conditional power computation, 6 through 9 are optional for unconditional power computation, and 10 is optional for power contour plotting.

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases in this job. If NCASES > 1, records 2, 3 and optional records must be repeated following the last record for each case.	1-2	I2
2	MSAMP	Sample size from population 1.	1-5	I5
	NSAMP	Sample size from population 2.	6-10	I5
3	NUMHYP	= 1, hypothesis (1) = 2, hypothesis (2) = 3, hypothesis (3)	1	I1
	ALPHA	Significance level α.	6-10	F5.4
	IGAMPU	=0, randomized test	15	I1
		=1, non-randomized test		
	ICP	=0, conditional power not requested.	20	I1
		=1, conditional power requested. Records 4 and 5 required.		
	IUCP	=0, unconditional power not requested.	25	I1
		=1, unconditional power requested. Records 6-9 required.		
	NPLOT	=0, conditional power plots not requested.	30	I1
		=1, conditional power plots requested. (Be careful. A "1" will generate $m+n$ plots.)		

Record types 4 and 5 required only if ICP = 1, i.e., conditional power requested.

4	NRHOS	Number of values of ρ. (NRHOS ≤ 32)	1-2	I2
5	VALRHO (I)	The values of ρ. As many as 4 records may be required. The ρ's should be entered in ascending order.	1-70	10F7.2

Record types 6-10 required only if IUCP = 1, i.e., unconditional power requested.

6	NP1S	Number of p_1's for unconditional power.	1-2	I2
	NP2S	Number of p_2's for unconditional power.	6-7	I2
7	P1(I)	The values of p_1. As many as 2 records may be required to enter the NP1S values.	1-80	16(2X, F3.2)
8	P2(I)	The values of p_2. As many as 2 records may be required to enter the NP2S values.	1-80	16(2X, F3.2)
9	NOPLOT	Number of equal probability power contours to plot. >0, record type 10 required. =0, contour power plots not requested. Record type 10 must be excluded. (NOPLOT ≤ 10)	1-5	I5
10	BETA(I)	The power values for the contour plots.	1-50	10F5.4

COMMENTS

As an aid to understanding the testing mechanism employed in BIN2POW, the sample space for the outcome of an experiment with $m = 8$ and $n = 10$ is shown in Figure 1. Also shown are the values of $X + Y = T$ for $T = 0,1,2,...,18$ and an arbitrary rejection region for a non-randomized test of hypothesis (1). The power of this test would be obtained by summing the joint probabilities in the rejection region for specific values of the p_i.

The concept of conditional power was not discussed in the PURPOSE section and will be addressed here. Conditional power of the test is the probability of rejecting the null hypothesis given knowledge of the number of successes in the combined sample from both populations. The conditional power is based on the conditional distribution of Y given $X + Y = t$. This probability distribution is as follows:

$$g_1(y \mid t) = \frac{\binom{m}{t-y}\binom{n}{y}\rho^y}{\sum_{i=\min y}^{\max y}\binom{m}{t-i}\binom{n}{i}\rho^i} \quad , \quad \min y \le y \le \max y$$

where $\min y = \max(0, t-m)$, $\max y = \min(t, n)$, and $\rho = p_2 q_1 / q_2 p_1$. Hence, for a given t and ρ, power is obtained by summing probabilities over the rejection region via $g_1(y \mid t)$.

Defining

$$CP_L(t,\rho) = \sum_{y=\min y}^{y_L - 1} g_1(y \mid t) + \gamma_L g_1(y_L \mid t)$$

and

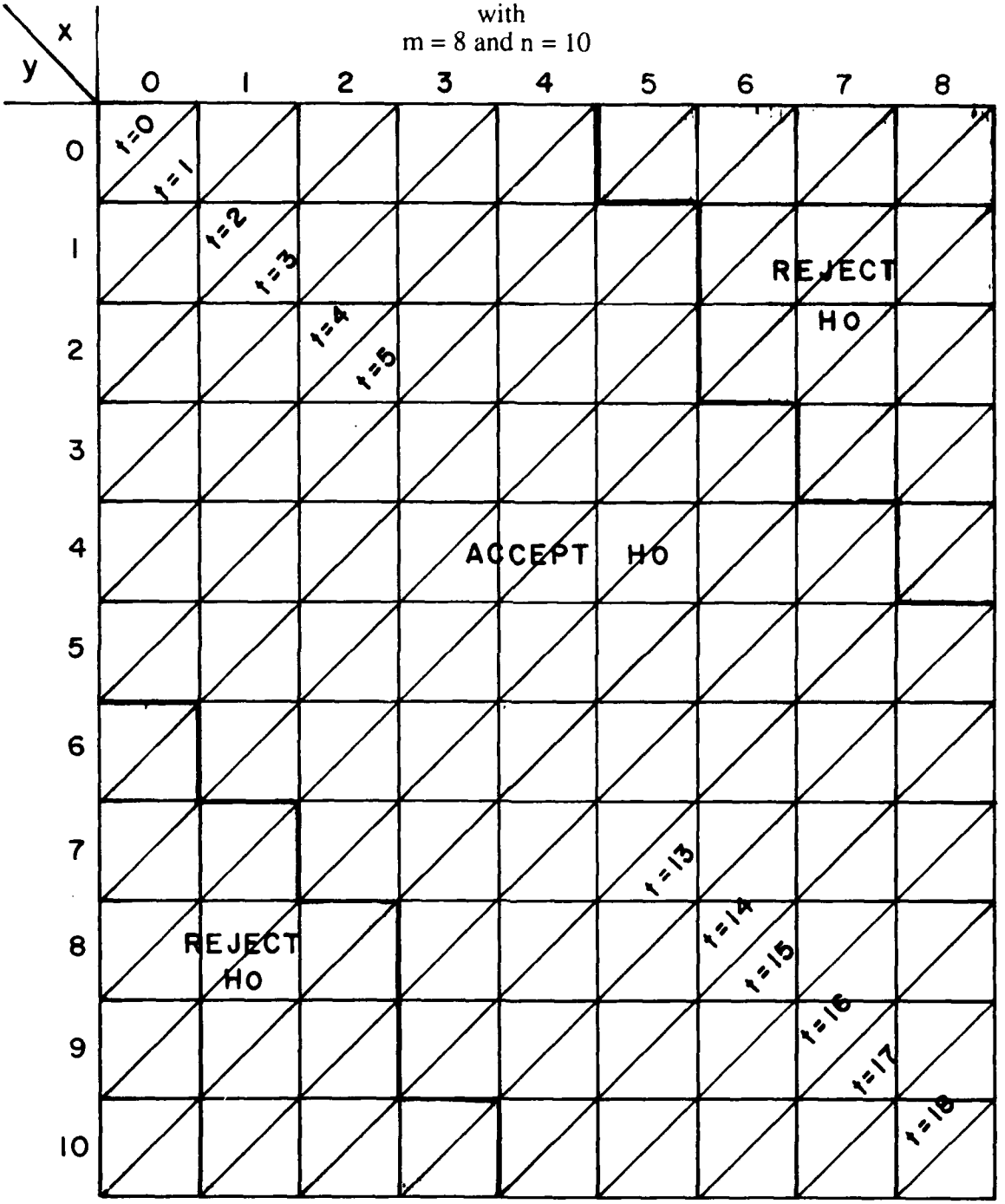
$$CP_U(t,\rho) = \sum_{y=y_U+1}^{\max y} g_1(y \mid t) + \gamma_U g_1(y_U \mid t)\,,$$

we can write conditional power as a function of $CP_L(t,\rho)$ and $CP_U(t,\rho)$ for each hypothesis as shown below:

Hypothesis	Conditional Power
1	$CP_L(t) + CP_U(t)$
2	$CP_L(t)$
3	$CP_U(t)$

BIN2POW has an option to plot the conditional power as a function of ρ (for up to 32 input values of ρ) for all $t = 0,1,\ldots,m+n$. One should use caution in exercising this option (NPLOT=1 on Record Type 3) since it provides a single plot for each t or a total of $m + n$ plots. If m or n is large, this will lead to the generation of considerable output.

SAMPLE SPACE
for
HYPOTHESIS 1
with
m = 8 and n = 10



Arbitrary rejection region for exemplification only. For $t = 6$, $y_L = 1$ and $y_U = 5$. For $t = 8$, $y_L = 3$ and $y_U = 6$, etc.

FIGURE 1

POI1POW

PURPOSE

Program POI1POW (Poisson Power, 1 Population) determines the critical (rejection) region and evaluates the power function for tests of hypotheses regarding the Poisson parameter λ. The tests are based on a random sample of size n from a Poisson population with probability distribution function

$$f(x) = e^{-\lambda}\lambda^x / x! \quad , \qquad \lambda > 0; \quad x = 0, 1, 2, \ldots$$

where x is the number of events in a time or space interval of length t (hereinafter referred to as a time unit) and λ is the mean rate or intensity, i.e., the expected number of events in a time unit is λ. The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are shown below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

T_0 = number of events in n time units (n time or space intervals of length t).

The probability distribution of T_0 depends on the hypothesized value of the parameter λ (λ_0 or $\lambda_1 \neq \lambda_0$). For $\lambda = \lambda_i$, this distribution will be denoted by $g(t_0 \mid \lambda_i)$ and has form

$$g(t_0 \mid \lambda_i) = e^{-n\lambda_i}(n\lambda_i)^{t_0} / t_0! \quad , \qquad t_0 = 0, 1, 2, \ldots$$

The critical region for the test is based on $g(t_0 \mid \lambda_0)$, and the power function is expressed in terms of $g(t_0 \mid \lambda_1)$.

* *

Hypothesis (1): H_0: $\lambda = \lambda_0$
H_1: $\lambda \neq \lambda_0$.

Critical Region: Reject H_0 if $T_0 < c_1$
Reject H_0 with probability γ_1 if $T_0 = c_1$
Reject H_0 if $T_0 > c_2$
Reject H_0 with probability γ_2 if $T_0 = c_2$
where c_1 is the largest integer such that

$$B_1 = \sum_{t_0=0}^{c_1-1} g(t_0 \mid \lambda_0) \quad \leq \quad \alpha/2$$

c_2 is the smallest integer such that

$$B_2 = \sum_{t_0=c_2+1}^{\infty} g(t_0 \mid \lambda_0) \quad \leq \quad \alpha/2$$

and

$\gamma_1 = (\alpha/2 - B_1)/g(c_1 \mid \lambda_0)$ (randomized test)

$\gamma_2 = (\alpha/2 - B_2)/g(c_2 \mid \lambda_0)$ (randomized test)

$\gamma_1 = \gamma_2 = 0$ (non-randomized test)

Power Function:

$$\Pi(\lambda_1, n) = \sum_{t_0=0}^{c_1-1} g(t_0 \mid \lambda_1) + \gamma_1 g(c_1 \mid \lambda_1)$$

$$+ \sum_{t_0=c_2+1}^{\infty} g(t_0 \mid \lambda_1) + \gamma_2 g(c_2 \mid \lambda_1) \quad .$$

* *

Hypothesis (2): H_0: $\lambda \leq \lambda_0$
H_1: $\lambda > \lambda_0$.

Critical Region: Reject H_0 if $T_0 > c$
Reject H_0 with probability γ if $T_0 = c$
where c is the smallest integer such that

$$A_1 = \sum_{t_0=c+1}^{\infty} g(t_0 \mid \lambda_0) \quad \leq \quad \alpha$$

and

$\gamma = (\alpha - A_1)/g(c \mid \lambda_0)$ (randomized test)

$= 0$ (non-randomized test)

Power Function:

$$\Pi(\lambda_1, n) = \sum_{t_0=c+1}^{\infty} g(t_0 \mid \lambda_1) + \gamma g(c \mid \lambda_1) \, .$$

* *

Hypothesis (3): $H_0: \lambda \geq \lambda_0$
$H_1: \lambda < \lambda_0$.

Critical Region: Reject H_0 if $T_0 < c$
Reject H_0 with probability γ if $T_0 = c$
where c is the largest integer such that

$$A_2 = \sum_{t_0=0}^{c-1} g(t_0 \mid \lambda_0) \quad \leq \quad \alpha$$

and

$\gamma = (\alpha - A_2) / g(c \mid \lambda_0)$ (randomized test)
$= 0$ (non-randomized test)

Power Function: $\Pi(\lambda_1, n) = \sum_{t_0=0}^{c-1} g(t_0 \mid \lambda_1) + \gamma g(c \mid \lambda_1)$.

FEATURES

POI1POW features include the following:

* Computation of the critical value c and randomized constant γ for up to 50 values of the sample size n and any one of the 3 hypotheses.
* Computation of power function for each value of n and up to 15 values of $\lambda = \lambda_1$.

REFERENCES

1. Feller, William (1960), ***An Introduction to Probability Theory and Its Applications***, John Wiley and Sons, Inc., pp. 146 - 154.
2. Krutchkoff, Richard G. (1970), ***Probability and Statistical Inference***, Gordon and Breach Science Publishers, pp. 224 - 229.

INPUT GUIDE

The user-created input file consists of two record types as described below.

Record Type	Variable	Description	Columns	Format
1	NHYP	= 1, hypothesis (1) = 2, hypothesis (2) = 3, hypothesis (3)	1-5	I5
	NTYPE	= 1, randomized test = 2, non-randomized test	6-10	I5
	NSS	Number of N(I) values (n values) to be read in on record type 2. (NSS ≤ 50)	11-15	I5
	K	Number of λ_1 values requested via $\lambda_1 = \lambda_0 \pm i*DELTA$, i=0,...,K for NHYP = 1 and via $\lambda_1 = \lambda_0 + i*DELTA$, i=0,...,K for NHYP = 2,3. K ≤ 7 for NHYP = 1. K ≤ 14 for NHYP = 2,3.	16-20	I5
	DELTA	Increment for generation of λ_1 values.	21-30	F10.5
	LAMDA	Hypothesized value of λ, i.e., λ_0	31-40	F10.5
	ALPHA	Significance level α.	41-50	F10.5
	LIMIT	N(I)*LAMDA > LIMIT activates the normal approximation to the Poisson. (DEFAULT = 1000.)	51-60	I10

Record type 2 contains the NSS N(I) values in format 10I8. As many as 5 type 2 records may be required.

CONTINUOUS

POWER

EVALUATIONS

NOR1PW

PURPOSE

Program NOR1PW (Normal Power, 1 Population) evaluates the power function for tests of hypotheses regarding the mean of a normal population. The tests are based on a random sample of size n from a normal population having known variance σ^2. Let X be a normal random variable with unknown mean μ and known variance σ^2. Then the sample mean $\overline{X}$ is a normal random variable with unknown mean μ and known variance σ^2/n, where $\overline{X} = \sum_{i=1}^{n} X_i/n$. These tests are commonly referred to as one-sample Z tests (see References 1 and 2). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = Z_0 = (\overline{X} - \mu_0) / (\sigma/\sqrt{n})$$

where μ_0 is the hypothesized value of the true mean μ. The probability density function of T_0 is normal with a variance of 1. Its mean depends on the true value of the parameter μ (μ_0 or $\mu_1 \neq \mu_0$). For $\mu = \mu_i$, this density will be denoted by $g(t_0 \mid \mu_i)$. Define $\delta = |\mu_i - \mu_0| / \sigma$. Then $g(t_0 \mid \mu_i)$ has the form

$$g(t_0 \mid \mu_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left[t_0 - \delta\sqrt{n}\right]^2} \quad , \qquad -\infty < t_0 < \infty$$

The critical region for the test is based on $g(t_0 \mid \mu_0)$ and the power function is expressed in terms of $g(t_0 \mid \mu_1)$.

* *

Hypothesis (1): H_0: $\mu = \mu_0$
H_1: $\mu \neq \mu_0$.

Critical Region: Reject H_0 if $|T_0| > z_{1-\alpha/2}$
where $z_{1-\alpha/2}$ is the 100(1 - α/2)th percentage point of the standard normal distribution (i.e., mean 0 and variance 1). Hence,

$$\alpha/2 = \int_{z_{1-\alpha/2}}^{\infty} g(t_0 \mid \mu_0) dt_0 .$$

Power Function: $\Pi(\mu_1, n) = \int_{-\infty}^{z_{\alpha/2}} g(t_0 \mid \mu_1) dt_0 + \int_{z_{1-\alpha/2}}^{\infty} g(t_0 \mid \mu_1) dt_0 .$

* *

Hypothesis (2): H_0: $\mu \leq \mu_0$
H_1: $\mu > \mu_0$.

Critical region: Reject H_0 if $T_0 > z_{1-\alpha}$
Hence,

$$\alpha = \int_{z_{1-\alpha}}^{\infty} g(t_0 \mid \mu_0) dt_0 .$$

Power Function: $\Pi(\mu_1, n) = \int_{z_{1-\alpha}}^{\infty} g(t_0 \mid \mu_1) dt_0 .$

* *

Hypothesis (3): H_0: $\mu \geq \mu_0$
H_1: $\mu < \mu_0$.

Critical Region: Reject H_0 if $T_0 < z_{\alpha}$
Hence,

$$\alpha = \int_{-\infty}^{z_{\alpha}} g(t_0 \mid \mu_0) dt_0 .$$

Power Function: $\Pi(\mu_1, n) = \int_{-\infty}^{z_{\alpha}} g(t_0 \mid \mu_1) dt_0 .$

FEATURES

NOR1PW allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. In NOR1PW the user specifies the type of test (i.e., one- or two-sided), and a range of values for sample size,

Type I error, and delta, where delta is the number of standard deviations that the true mean is from the hypothesized mean in absolute value. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. Each page of NOR1PW output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of n and up to 9 values of delta.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 183 - 198.
2. Brownlee, K. A. (1965), ***Statistical Theory and Methodology in Science and Engineering***, Second Edition, John Wiley and Sons, Inc., pp. 105 - 109 and pp. 113 - 118.
3. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 262 - 263.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the limits for delta, sample size, and Type I error together with the test type. If NCASES is greater than one, record type 2 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5
2	DELTALL	Lower limit of delta values (DELTALL ≥ 0.0)	1 - 8	F8.0
	DINCR	Delta value increment (DINCR > 0.0)	9 - 16	F8.0
	DELTAUL	Upper limit of delta values (DELTAUL ≥ 0.0)	17 - 24	F8.0
	FNLL	Lower limit of sample size values (FNLL > 0)	25 - 32	F8.0

FNINCR	Sample size value increment (FINCR > 0)	33 - 40	F8.0
FNUL	Upper limit of sample size values (FNUL > 0)	41 - 48	F8.0
ALPHALL	Lower limit of Type I error values (0.0 < ALPHALL < 1.0)	49 - 56	F8.0
AINCR	Type I error value increment (AINCR > 0.0)	57 - 64	F8.0
ALPHAUL	Upper limit of Type I error values (0.0 < ALPHAUL < 1.0)	65 - 72	F8.0
TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	73 - 80	F8.0

One record type 2 is required for each case to be processed.

NOR2PWE

PURPOSE

Program NOR2PWE (Normal Power, 2 Populations, Equal Sample Sizes) evaluates the power function for tests of hypotheses regarding the equality of the means of two normal populations. The tests are based on two independent random samples each of size n from two normal populations having known variances σ_1^2 and σ_2^2. Let X_1 and X_2 be normal random variables with unknown means μ_1 and μ_2 and known variances σ_1^2 and σ_2^2, respectively. Then the sample means $\overline{X}_1$ and $\overline{X}_2$ are normal random variables with unknown means μ_1 and μ_2 and known variances σ_1^2/n and σ_2^2/n, respectively, where $\overline{X}_i = \sum_{j=1}^{n} X_{ij}/n_i$. These tests are commonly referred to as two-sample Z tests (see Reference 1). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = Z_0 = (\overline{X}_1 - \overline{X}_2) / \sqrt{(\sigma_1^2 + \sigma_2^2)/n} \, .$$

The probability density function of T_0 is normal with a variance of 1. Its mean depends on the true value of the difference $d = \mu_1 - \mu_2$ ($d = d_0$ or $d = d_1 \neq d_0$). $d_0 = 0$ is the hypothesized value of d. For $d = d_i$, this density will be denoted by $g(t_0 \mid d_i)$. Define $\delta = |d_i| / \sqrt{\sigma_1^2 + \sigma_2^2}$. Then $g(t_0 \mid d_i)$ has the form

$$g(t_0 \mid d_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left[t_0 - \delta\sqrt{n}\right]^2} \quad , \qquad -\infty < t_0 < \infty$$

The critical region for the test is based on $g(t_0 \mid d_0)$ and the power function is expressed in terms of $g(t_0 \mid d_1)$.

* *

Hypothesis (1): H_0: $\mu_1 = \mu_2$
H_1: $\mu_1 \neq \mu_2$.

Critical Region: Reject H_0 if $|T_0| > z_{1-\alpha/2}$

where $z_{1-\alpha/2}$ is the 100(1 - $\alpha/2$)th percentage point of the standard normal distribution (i.e., mean 0 and variance 1). Hence,

$$\alpha/2 = \int_{z_{1-\alpha/2}}^{\infty} g(t_0 \mid d_0) dt_0 .$$

Power Function: $\prod(d_1, n) = \int_{-\infty}^{z_{\alpha/2}} g(t_0 \mid d_1) dt_0 + \int_{z_{1-\alpha/2}}^{\infty} g(t_0 \mid d_1) dt_0 .$

* *

Hypothesis (2): H_0: $\mu_1 \leq \mu_2$

H_1: $\mu_1 > \mu_2$.

Critical region: Reject H_0 if $T_0 > z_{1-\alpha}$

Hence,

$$\alpha = \int_{z_{1-\alpha}}^{\infty} g(t_0 \mid d_0) dt_0 .$$

Power Function: $\prod(d_1, n) = \int_{z_{1-\alpha}}^{\infty} g(t_0 \mid d_1) dt_0 .$

* *

Hypothesis (3): H_0: $\mu_1 \geq \mu_2$

H_1: $\mu_1 < \mu_2$.

Critical Region: Reject H_0 if $T_0 < z_{\alpha}$

Hence,

$$\alpha = \int_{-\infty}^{z_{\alpha}} g(t_0 \mid d_0) dt_0 .$$

Power Function: $\prod(d_1, n) = \int_{-\infty}^{z_{\alpha}} g(t_0 \mid d_1) dt_0 .$

FEATURES

NOR2PWE allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. In NOR2PWE the user specifies the type of test (i.e., one- or two-sided), and a range of values for sample size, Type I error, and delta, where delta is the number of standard deviations that the true mean is from the hypothesized mean in absolute value. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. Each page of NOR2PWE output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of n and up to 9 values of delta.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc.. pp. 225 - 235.
2. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 266 - 267.
3. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, pp. 283 - 284.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the limits for delta, sample size, and Type I error together with the test type. If NCASES is greater than one, record type 2 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5
2	DELTALL	Lower limit of delta values (DELTALL $\geq$ 0.0)	1 - 8	F8.0
	DINCR	Delta value increment (DINCR > 0.0)	9 - 16	F8.0
	DELTAUL	Upper limit of delta values (DELTAUL $\geq$ 0.0)	17 - 24	F8.0

FNLL	Lower limit of sample size values (FNLL > 0)	25 - 32	F8.0
FNINCR	Sample size value increment (FINCR > 0)	33 - 40	F8.0
FNUL	Upper limit of sample size values (FNUL > 0)	41 - 48	F8.0
ALPHALL	Lower limit of Type I error values (0.0 < ALPHALL < 1.0)	49 - 56	F8.0
AINCR	Type I error value increment (AINCR > 0.0)	57 - 64	F8.0
ALPHAUL	Upper limit of Type I error values (0.0 < ALPHAUL < 1.0)	65 - 72	F8.0
TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	73 - 80	F8.0

One record type 2 is required for each case to be processed.

NOR2PWU

PURPOSE

Program NOR2PWU (Normal Power, 2 Populations, Unequal Sample Sizes) evaluates the power function for tests of hypotheses regarding the equality of the means of two normal populations. The tests are based on two independent random samples each of sizes n_1 and n_2, respectively, from two normal populations having known variances σ_1^2 and σ_2^2. If the sample sizes are equal, program NOR2PWE should be used. Let X_1 and X_2 be normal random variables with unknown means μ_1 and μ_2 and known variances σ_1^2 and σ_2^2, respectively. Then the sample means $\overline{X}_1$ and $\overline{X}_2$ are normal random variables with unknown means μ_1 and μ_2 and known variances σ_1^2/n_1 and σ_2^2/n_2, respectively, where $\overline{X}_i = \sum_{j=1}^{n} X_{ij}/n_i$. These tests are commonly referred to as two-sample Z tests (see Reference 1). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = Z_0 = (\overline{X}_1 - \overline{X}_2) / \sqrt{(\sigma_1^2/n_1 + \sigma_2^2/n_2)}\,.$$

The probability density function of T_0 is normal with a variance of 1. Its mean depends on the true value of the difference $d = \mu_1 - \mu_2$ ($d = d_0$ or $d = d_1 \neq d_0$). $d_0 = 0$ is the hypothesized value of d. For $d = d_i$, this density will be denoted by $g(t_0 \mid d_i)$. Define $\delta = |d_i|$. Then $g(t_0 \mid d_i)$ has the form

$$g(t_0 \mid d_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left[t_0 - \delta/\sqrt{(\sigma_1^2/n_1 + \sigma_2^2/n_2)}\right]^2}, \qquad -\infty < t_0 < \infty$$

The critical region for the test is based on $g(t_0 \mid d_0)$ and the power function is expressed in terms of $g(t_0 \mid d_1)$.

* *

Hypothesis (1): H_0: $\mu_1 = \mu_2$
H_1: $\mu_1 \neq \mu_2$.

Critical Region: Reject H_0 if $|T_0| > z_{1-\alpha/2}$
where $z_{1-\alpha/2}$ is the $100(1 - \alpha/2)$th percentage point of the standard normal distribution (i.e., mean 0 and variance 1). Hence,

$$\alpha/2 = \int_{z_{1-\alpha/2}}^{\infty} g(t_0 \mid d_0)dt_0 .$$

Power Function: $\prod(d_1, n_1, n_2) = \int_{-\infty}^{z_{\alpha/2}} g(t_0 \mid d_1)dt_0 + \int_{z_{1-\alpha/2}}^{\infty} g(t_0 \mid d_1)dt_0 .$

* *

Hypothesis (2): H_0: $\mu_1 \leq \mu_2$
H_1: $\mu_1 > \mu_2$.

Critical region: Reject H_0 if $T_0 > z_{1-\alpha}$
Hence,

$$\alpha = \int_{z_{1-\alpha}}^{\infty} g(t_0 \mid d_0)dt_0 .$$

Power Function: $\prod(d_1, n_1, n_2) = \int_{z_{1-\alpha}}^{\infty} g(t_0 \mid d_1)dt_0 .$

* *

Hypothesis (3): H_0: $\mu_1 \geq \mu_2$
H_1: $\mu_1 < \mu_2$.

Critical Region: Reject H_0 if $T_0 < z_{\alpha}$
Hence,

$$\alpha = \int_{-\infty}^{z_{\alpha}} g(t_0 \mid d_0)dt_0 .$$

Power Function: $\prod(d_1, n_1, n_2) = \int_{-\infty}^{z_{\alpha}} g(t_0 \mid d_1)dt_0 .$

FEATURES

NOR2PWU allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. NOR2PWU provides the user with the option of either computing a single power for a specified sample size pair or printing a table of power values. The known values of the population standard deviations are always required. If the single power option is chosen, the user must additionally specify the type of test (i.e., one- or two-sided), a sample size pair, and a value of delta, where delta is the absolute difference in the true means. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. If the power table option is chosen, the user instead specifies the test type and a range of values for the first population sample size, Type I error, and delta. In addition, a multiplicative factor must be provided which relates the second population sample size to the first. Under the power table option each page of NOR2PWU output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of the pair (n_1, n_2) and up to 9 values of delta.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 225 - 235.
2. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 266 - 267.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the known population standard deviations. The third record type specifies the power table computation option. If the single power computation option is chosen, the fourth record type is different and a fifth record type is also required. In this case, the fourth record type specifies the limits for delta, the first population sample size, and the Type I error together with the test type. The fifth record type specifies the value of the multiplicative factor relating the second population sample size to the first. If NCASES is greater than one, record types 2, 3, 4 (4A or 4B) and, if necessary, 5 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5

2	SIGMA1	1st population standard deviation (SIGMA1 > 0.0)	1-20	F20.0
	SIGMA2	2nd population standard deviation (SIGMA2 > 0.0)	21-40	F20.0
3	IOPTION	Power computation option = 0, for power table output = 1, for single power value	1 - 5	I5

The following record type is included only if IOPTION = 1.

4A	D(1)	Delta value (D(1) ≥ 0.0)	1 - 8	F8.0
	FN(1)	1st population sample size (FN(1) > 0)	9 - 16	F8.0
	FN2	2nd population sample size (FN2 > 0)	17 - 24	F8.0
	ALPH(1)	Type I error (0.0 < ALPH(1) < 1.00)	25 - 32	F8.0
	TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	33 - 40	F8.0

The following record type is included only if IOPTION = 0.

4B	DELTALL	Lower limit of delta values (DELTALL ≥ 0.0)	1 - 8	F8.0
	DINCR	Delta value increment (DINCR > 0.0)	9 - 16	F8.0
	DELTAUL	Upper limit of delta values (DELTAUL ≥ 0.0)	17 - 24	F8.0
	FNLL	Lower limit of first population sample size values (FNLL > 0)	25 - 32	F8.0

	FNINCR	First population sample size value increment	33 - 40	F8.0
		(FNINCR > 0)		
	FNUL	Upper limit of first population sample size values	41 - 48	F8.0
		(FNUL > 0)		
	ALPHALL	Lower limit of Type I error values	49 - 56	F8.0
		(0.0 < ALPHALL < 1.0)		
	AINCR	Type I error value increment	57 - 64	F8.0
		(AINCR > 0.0)		
	ALPHAUL	Upper limit of Type I error values	65 - 72	F8.0
		(0.0 < ALPHAUL < 1.0)		
	TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	73 - 80	F8.0
		Record type 5 is included only if IOPTION = 0.		
5	FACTOR	Sample size multiplicative factor. Satisfies the relationship n_2 = FACTOR*n_1.	1-20	F20.0
		(FACTOR > 0.0)		

One record type 2, 3, and 4 (4A or 4B) is required for each case to be processed. One record type 5 is required for each cased in which IOPTION = 0.

T1POW

PURPOSE

Program T1POW (T Power, 1 Population) evaluates the power function for tests of hypotheses regarding the mean of a normal population. The tests are based on a random sample of size n from a normal population having unknown variance σ^2. Let X be a normal random variable with unknown mean μ and unknown variance σ^2/n. Then the sample mean $\overline{X}$ is a normal random variable with unknown mean μ and unknown variance σ^2/n, where $\overline{X} = \sum_{i=1}^{n} X_i/n$. These tests are commonly referred to as one-sample t tests or small-sample t tests (see Reference 1). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = t_0 = (\overline{X} - \mu_0)/(S/\sqrt{n})$$

where μ_0 is the hypothesized value of the true mean μ. S is the positive square root of S^2, the unbiased estimator of σ^2, where $S^2 = \sum_{i=1}^{n}(X_i - \overline{X})^2/(n-1)$. The probability density function of T_0 depends on the true value of the parameter μ (μ_0 or $\mu_1 \neq \mu_0$). For $\mu = \mu_i$, this density will be denoted by $g(t_0 \mid \mu_i)$. Define $\delta = |\mu_i - \mu_0|/\sigma$. Then $g(t_0 \mid \mu_i)$ is a non-central t distribution with parameters ν (degrees of freedom) and $\delta' = \delta/\sqrt{n}$ (non-centrality parameter) and is given by

$$g(t_0 \mid \mu_i) = \frac{\left(\frac{\nu}{2}\right)^{\frac{\nu}{2}} e^{-(\delta')^2/2}}{\sqrt{2\pi}\,\Gamma\left(\frac{\nu}{2}\right)} \sum_{r=0}^{\infty} \frac{(t_0\delta')^r}{r!} \Gamma\left[\frac{1}{2}(\nu + r + 1)\right] \left(\frac{2}{\nu + t_0^2}\right)^{\frac{1}{2}(\nu+r+1)}, \quad -\infty < t_0 < \infty$$

Here $\nu = n - 1$. When $\mu_i = \mu_0$, $\delta' = 0$ and $g(t_0 \mid \mu_i)$ reduces to the central t distribution with $n - 1$ degrees of freedom. Otherwise, δ' is non-zero. The critical region for the test is based on $g(t_0 \mid \mu_0)$ and the power function is expressed in terms of $g(t_0 \mid \mu_1)$.

* *

Hypothesis (1): H_0: $\mu = \mu_0$
H_1: $\mu \neq \mu_0$.

Critical Region: Reject H_0 if $|T_0| > t_{\nu,1-\alpha/2}$

where $t_{\nu,1-\alpha/2}$ is the 100(1 - $\alpha/2$)th percentage point of the central t distribution with ν degrees of freedom. Hence,

$$\alpha/2 = \int_{t_{\nu,1-\alpha/2}}^{\infty} g(t_0 \mid \mu_0) dt_0 .$$

Power Function: $$\prod(\mu_1, n) = \int_{-\infty}^{t_{\nu,\alpha/2}} g(t_0 \mid \mu_1) dt_0 + \int_{t_{\nu,1-\alpha/2}}^{\infty} g(t_0 \mid \mu_1) dt_0 .$$

* *

Hypothesis (2): H_0: $\mu \leq \mu_0$

H_1: $\mu > \mu_0$.

Critical region: Reject H_0 if $T_0 > t_{\nu,1-\alpha}$

Hence,

$$\alpha = \int_{t_{\nu,1-\alpha}}^{\infty} g(t_0 \mid \mu_0) dt_0 .$$

Power Function: $$\prod(\mu_1, n) = \int_{t_{\nu,1-\alpha}}^{\infty} g(t_0 \mid \mu_1) dt_0 .$$

* *

Hypothesis (3): H_0: $\mu \geq \mu_0$

H_1: $\mu < \mu_0$.

Critical Region: Reject H_0 if $T_0 < t_{\nu,\alpha}$

Hence,

$$\alpha = \int_{-\infty}^{t_{\nu,\alpha}} g(t_0 \mid \mu_0) dt_0 .$$

Power Function: $$\prod(\mu_1, n) = \int_{-\infty}^{t_{\nu,\alpha}} g(t_0 \mid \mu_1) dt_0 .$$

FEATURES

T1POW allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. In T1POW the user specifies the type of test (i.e., one- or two-sided), and a range of values for sample size, Type I error, and delta, where delta is the number of standard deviations that the true mean is from the hypothesized mean in absolute value. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. Each page of T1POW output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of *n* and up to 9 values of delta.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 198 - 207.
2. Brownlee, K. A. (1965), ***Statistical Theory and Methodology in Science and Engineering***, Second Edition, John Wiley and Sons, Inc., pp. 295 - 296.
3. DiDonato, A. R. (1988), unpublished notes and computer code for numerically evaluating a specialized infinite sum occurring in the expression for the power of the t test.
4. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 263 - 264.
5. Thomas, M. A. (1988), unpublished notes deriving an expression for the power of the t test in terms of the incomplete beta function.
6. Wine, R. L. (1964), ***Statistics for Scientists and Engineers***, Prentice-Hall, Inc., pp. 254 - 260.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the limits for delta, the sample size, and Type I error together with the test type. If NCASES is greater than one, record type 2 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5
2	DELTALL	Lower limit of delta values (DELTALL ≥ 0.0)	1 - 8	F8.0
	DINCR	Delta value increment (DINCR > 0.0)	9 - 16	F8.0
	DELTAUL	Upper limit of delta values (DELTAUL ≥ 0.0)	17 - 24	F8.0
	FNLL	Lower limit of sample size values (FNLL > 0)	25 - 32	F8.0
	FNINCR	Sample size value increment (FNINCR > 0)	33 - 40	F8.0
	FNUL	Upper limit of sample size values (FNUL > 0)	41 - 48	F8.0
	ALPHALL	Lower limit of Type I error values (0.0 < ALPHALL < 1.0)	49 - 56	F8.0
	AINCR	Type I error value increment (AINCR > 0.0)	57 - 64	F8.0
	ALPHAUL	Upper limit of Type I error values (0.0 < ALPHAUL < 1.0)	65 - 72	F8.0
	TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	73 - 80	F8.0

One record type 2 is required for each case to be processed.

COMMENTS

Since the integrals in the expressions for power cannot be evaluated in closed form, they must be evaluated numerically. For each of the three hypotheses it can be shown that the power expression can be transformed to an expression involving the incomplete beta function $I_x(a,b)$, where

$$I_x(a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^x t^{a-1}(1-t)^{b-1}dt \qquad \begin{matrix} 0 \le x \le 1 \\ a,b > 0 \end{matrix}$$

Very good numerical routines are available for evaluating $I_x(a,b)$ such as BRATIO in MATHLIB. Transforming the power expressions results in two expressions for power, one for the one-sided test (hypothesis (2) or (3)) and one for the two-sided test (hypothesis (1)). These expressions are given below:

(1) One-sided test

$$\text{Power} = 0.5[e^{-(\delta')^2/2} \sum_{i=0}^{\infty} \frac{(\delta'^2/2)^i}{\Gamma(i+1)} I_{1-x}(\nu/2, i+0.5)$$

$$+ e^{-(\delta')^2/2} \sum_{i=0}^{\infty} \frac{(\delta'^2/2)^{i+1/2}}{\Gamma(i+3/2)} I_{1-x}(\nu/2, i+1)]$$

(2) Two-sided test

$$\text{Power} = e^{-(\delta')^2/2} \sum_{i=0}^{\infty} \frac{(\delta'^2/2)^i}{\Gamma(i+1)} I_{1-x}(\nu/2, i+0.5)$$

Note that $I_{1-x}(a,b) = 1 - I_x(b,a)$. Program T1POW utilizes the above expressions with $\nu = n-1$ and $\delta' = \delta\sqrt{n}$ to compute the power of the one-sample t test.

T2POW

PURPOSE

Program T2POW (T Power, 2 Populations) evaluates the power function for tests of hypotheses regarding the equality of the means of two normal populations. The tests are based on two independent random samples of sizes n_1 and n_2, respectively, from two normal populations having unknown but equal variances. Let X_1 and X_2 be normal random variables with unknown means μ_1 and μ_2 and unknown common variance σ^2. Then the sample means $\overline{X}_1$ and $\overline{X}_2$ are normal random variables with unknown means μ_1 and μ_2 and unknown variances σ^2/n_1 and σ^2/n_2, respectively, where $\overline{X}_i = \sum_{j=1}^{n_i} X_{ij}/n_i$. These tests are commonly referred to as two-sample t tests or "pooled" t tests (see Reference 1). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = t_0 = (\overline{X}_1 - \overline{X}_2) / S_p\sqrt{1/n_1 + 1/n_2}$$

where S_p^2 is the "pooled" estimator of the common variance σ^2 and is given by

$$S_p^2 = \left[\sum_{j=1}^{n_1} (X_{1j} - \overline{X}_1)^2 + \sum_{j=1}^{n_2} (X_{2j} - \overline{X}_2)^2 \right] / (n_1 + n_2 - 2)$$

The probability density function of T_0 depends on the true value of the difference $d = \mu_1 - \mu_2$ ($d = d_0$ or $d = d_1 \neq d_0$). $d_0 = 0$ is the hypothesized value of d. For $d = d_i$, this density will be denoted by $g(t_0 \mid d_i)$. Define $\delta = |d_i|/\sigma$. Then $g(t_0 \mid d_i)$ is a non-central t distribution with parameters ν (degrees of freedom) and $\delta' = \delta/\sqrt{1/n_1 + 1/n_2}$ (non-centrality parameter) and is given by

$$g(t_0 \mid d_i) = \frac{\left(\frac{\nu}{2}\right)^{\frac{\nu}{2}} e^{-(\delta')^2/2}}{\sqrt{2\pi}\,\Gamma\left(\frac{\nu}{2}\right)} \sum_{r=0}^{\infty} \frac{(t_0\delta')^r}{r!} \Gamma\left[\frac{1}{2}(\nu + r + 1)\right] \left(\frac{2}{\nu + t_0^2}\right)^{\frac{1}{2}(\nu + r + 1)}, \quad -\infty < t_0 < \infty$$

Here $\nu = n_1 + n_2 - 2$. When $d_i = d_0 = 0$, $\delta' = 0$ and $g(t_0 \mid d_i)$ reduces to the central t distribution with $n_1 + n_2 - 2$ degrees of freedom. Otherwise, δ' is non-zero. The critical region for the test is based on $g(t_0 \mid d_0)$ and the power function is expressed in terms of $g(t_0 \mid d_1)$.

* *

Hypothesis (1): H_0: $\mu_1 = \mu_2$

H_1: $\mu_1 \neq \mu_2$.

Critical Region: Reject H_0 if $|T_0| > t_{\nu,1-\alpha/2}$

where $t_{\nu,1-\alpha/2}$ is the 100(1 - $\alpha/2$)th percentage point of the central t distribution with ν degrees of freedom. Hence,

$$\alpha/2 = \int_{t_{\nu,1-\alpha/2}}^{\infty} g(t_0 \mid d_0)dt_0 .$$

Power Function: $\prod(d_1, n_1, n_2) = \int_{-\infty}^{t_{\nu,\alpha/2}} g(t_0 \mid d_1)dt_0 + \int_{t_{\nu,1-\alpha/2}}^{\infty} g(t_0 \mid d_1)dt_0 .$

* *

Hypothesis (2): H_0: $\mu_1 \leq \mu_2$

H_1: $\mu_1 > \mu_2$

Critical region: Reject H_0 if $T_0 > t_{\nu,1-\alpha}$

Hence,

$$\alpha = \int_{t_{\nu,1-\alpha}}^{\infty} g(t_0 \mid d_0)dt_0 .$$

Power Function: $\prod(d_1, n_1, n_2) = \int_{t_{\nu,1-\alpha}}^{\infty} g(t_0 \mid d_1)dt_0 .$

* *

Hypothesis (3): H_0: $\mu_1 \geq \mu_2$

H_1: $\mu_1 < \mu_2$

Critical Region: Reject H_0 if $T_0 < t_{\nu,\alpha}$

Hence,

$$\alpha = \int_{-\infty}^{t_{\nu,\alpha}} g(t_0 \mid d_0)dt_0 .$$

Power Function: $\Pi(d_1, n_1, n_2) = \int_{-\infty}^{t_{\nu,\alpha}} g(t_0 \mid d_1) dt_0$.

FEATURES

T2POW allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. T2POW provides the user with the option of either computing a single power for a specified sample size pair or printing a table of power values. If the single power option is chosen, the user must also specify the type of test (i.e., one- or two-sided), a sample size pair, and a value of delta, where delta is the number of standard deviations that the difference in true means is from the hypothesized difference of zero in absolute value. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. If the power table option is chosen, the user instead specifies the test type, and a range of values for the first population sample size, Type I error, and delta. In addition, a multiplicative factor must be provided which relates the second population sample size to the first. Under the power table option each page of T2POW output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of the pair (n_1, n_2) and up to 9 values of delta.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 235 - 240.
2. Brownlee, K. A. (1965), ***Statistical Theory and Methodology in Science and Engineering***, Second Edition, John Wiley and Sons, Inc., pp. 297 - 299.
3. DiDonato, A. R. (1988), unpublished notes and computer code for numerically evaluating a specialized infinite sum occurring in the expression for the power of the t test.
4. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 267 - 269.
5. Thomas, M. A. (1988), unpublished notes deriving an expression for the power of the t test in terms of the incomplete beta function.
6. Wine, R. L. (1964), ***Statistics for Scientists and Engineers***, Prentice-Hall, Inc., pp. 254 - 264.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the power table computation option. If the single power computation option is chosen, the third record type specifies the values of delta, the first population sample size, the second population sample size, the Type I error, and the test type. If the power table option is chosen, the third record type is different and a fourth record type is also required. In this case, the third record type specifies the limits for delta, the first population sample size, and Type I error together with the test type. The fourth record type specifies the value of the multiplicative factor relating the second population sample size to the first. If NCASES is greater than one, record types 2, 3 (3A or 3B), and, if necessary, 4 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5
2	IOPTION	Power computation option = 0, for power table output = 1, for single power value	1 - 5	I5
		The following record type is included only if IOPTION = 1.		
3A	D(1)	Delta value $(D(1) \geq 0.0)$	1 - 8	F8.0
	FN(1)	1st population sample size $(FN(1) > 0)$	9 - 16	F8.0
	FN2	2nd population sample size $(FN2 > 0)$	17 - 24	F8.0
	ALPH(1)	Type I error $(0.0 < ALPH(1) < 1.00)$	25 - 32	F8.0
	TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	33 - 40	F8.0

The following record type is included only if IOPTION = 0.

3B	DELTALL	Lower limit of delta values (DELTALL ≥ 0.0)	1 - 8	F8.0
	DINCR	Delta value increment (DINCR > 0.0)	9 - 16	F8.0
	DELTAUL	Upper limit of delta values (DELTAUL ≥ 0.0)	17 - 24	F8.0
	FNLL	Lower limit of first population sample size values (FNLL > 0)	25 - 32	F8.0
	FNINCR	First population sample size value increment (FINCR > 0)	33 - 40	F8.0
	FNUL	Upper limit of first population sample size values (FNUL > 0)	41 - 48	F8.0
	ALPHALL	Lower limit of Type I error values (0.0 < ALPHALL < 1.0)	49 - 56	F8.0
	AINCR	Type I error value increment (AINCR > 0.0)	57 - 64	F8.0
	ALPHAUL	Upper limit of Type I error values (0.0 < ALPHAUL < 1.0)	65 - 72	F8.0
	TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	73 - 80	F8.0

Record type 4 is included only if IOPTION = 0.

4	FACTOR	Sample size multiplicative factor. Satisfies the relationship n_2 = FACTOR*n_1. (FACTOR > 0.0)	1 - 20	F8.0

One record type 2 and 3 (3A or 3B) is required for each case to be processed. One record type 4 is required for each case in which IOPTION = 0.

COMMENTS

Since the integrals in the expressions for power cannot be evaluated in closed form, they must be evaluated numerically. For each of the three hypotheses it can be shown that the power expression can be transformed to an expression involving the incomplete beta function $I_x(a,b)$, where

$$I_x(a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^x t^{a-1}(1-t)^{b-1}dt \qquad \begin{matrix} 0 \le x \le 1 \\ a,b > 0 \end{matrix}$$

Very good numerical routines are available for evaluating $I_x(a,b)$ such as BRATIO in MATHLIB. Transforming the power expressions results in two expressions for power, one for the one-sided test (hypothesis (2) or (3)) and one for the two-sided test (hypothesis (1)). These expressions are given below:

(1) <u>One-sided test</u>

$$\text{Power} = 0.5[e^{-(\delta')^2/2} \sum_{i=0}^{\infty} \frac{(\delta'^2/2)^i}{\Gamma(i+1)} I_{1-x}(\nu/2, i+0.5)$$

$$+ e^{-(\delta')^2/2} \sum_{i=0}^{\infty} \frac{(\delta'^2/2)^{i+1/2}}{\Gamma(i+3/2)} I_{1-x}(\nu/2, i+1)]$$

(2) <u>Two-sided test</u>

$$\text{Power} = e^{-(\delta')^2/2} \sum_{i=0}^{\infty} \frac{(\delta'^2/2)^i}{\Gamma(i+1)} I_{1-x}(\nu/2, i+0.5)$$

Note that $I_{1-x}(a,b) = 1 - I_x(b,a)$. Program T2POW utilizes the above expressions with $\nu = n_1 + n_2 - 2$ and $\delta' = \delta/\sqrt{1/n_1 + 1/n_2}$ to compute the power of the pooled t test.

CHIVPOW

PURPOSE

Program CHIVPOW (Power of the Chi-Square Test on a Single Variance) evaluates the power function for tests of hypotheses regarding the variance of a normal population. The tests are based on a random sample of size n from a normal population having unknown variance σ^2. Let X be a normal random variable with unknown mean μ and unknown variance σ^2. Then the sample mean $\overline{X}$ is a normal random variable with unknown mean μ and unknown variance σ^2/n, where $\overline{X} = \sum_{i=1}^{n} X_i/n$. These tests are commonly referred to as χ^2 tests on a single variance (see References 1 and 2). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = \chi_0^2 = \sum_{i=1}^{n} (X_i - \overline{X})^2 / \sigma_0^2$$

where σ_0^2 is the hypothesized value of the true variance σ^2. The probability density function of T_0 depends on the true value of the parameter σ (σ_0 or $\sigma_1 \neq \sigma_0$). For $\sigma = \sigma_0$, this density will be denoted by $g(t_0 \mid \sigma_0)$. $g(t_0 \mid \sigma_0)$ is a chi-square distribution with ν (degrees of freedom) and is given by

$$g(t_0 \mid \sigma_0) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)} t_0^{\frac{\nu}{2}-1} e^{-t_0/2}, \quad t_0 > 0 \quad ,$$

Here $\nu = n - 1$. Define $\lambda = \sigma_i/\sigma_0$ and $c = 1/\lambda$. λ ranges from 0 to ∞ and has a value of 1 when the hypothesized value of the population standard deviation equals the true value of that parameter. The critical region for the test is based on $g(t_0 \mid \sigma_0)$ and the power function is expressed in terms of λ and $g(t_0 \mid \sigma_0)$.

* *

Hypothesis (1): H_0: $\sigma = \sigma_0$

H_1: $\sigma \neq \sigma_0$.

Critical Region: Reject H_0 if $T_0 < \chi^2_{\nu,\alpha/2}$ or if $T_0 > \chi^2_{\nu,1-\alpha/2}$

where $\chi^2_{\nu,1-\alpha/2}$ is the 100(1 - α/2)th percentage point of the chi-square distribution with ν degrees of freedom. Hence,

$$\alpha/2 = \int_{\chi^2_{\nu,1-\alpha/2}}^{\infty} g(t_0 \mid \sigma_0) dt_0 .$$

Power Function: $$\prod(\sigma_1, n) = \int_0^{c^2\chi^2_{\nu,\alpha/2}} g(t_0 \mid \sigma_0) dt_0 + \int_{c^2\chi^2_{\nu,1-\alpha/2}}^{\infty} g(t_0 \mid \sigma_0) dt_0 .$$

* *

Hypothesis (2): H_0: $\sigma \le \sigma_0$

H_1: $\sigma > \sigma_0$.

Critical region: Reject H_0 if $T_0 > \chi^2_{\nu,1-\alpha}$

Hence,

$$\alpha = \int_{\chi^2_{\nu,1-\alpha}}^{\infty} g(t_0 \mid \sigma_0) dt_0 .$$

Power Function: $$\prod(\sigma_1, n) = \int_{c^2\chi^2_{\nu,1-\alpha}}^{\infty} g(t_0 \mid \sigma_0) dt_0 .$$

* *

Hypothesis (3): H_0: $\sigma \ge \sigma_0$

H_1: $\sigma < \sigma_0$.

Critical Region: Reject H_0 if $T_0 < \chi^2_{\nu,\alpha}$

Hence,

$$\alpha = \int_0^{\chi^2_{\nu,\alpha}} g(t_0 \mid \sigma_0) dt_0 .$$

Power Function: $$\prod(\sigma_1, n) = \int_0^{c^2\chi^2_{\nu,\alpha}} g(t_0 \mid \sigma_0) dt_0 .$$

FEATURES

CHIVPOW allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. In CHIVPOW the user specifies the type of test (i.e., one- or two-sided), and a range of values for sample size, Type I error, and lambda, where lambda is the ratio of the true population standard deviation to its hypothesized value. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. Each page of CHIVPOW output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of n and up to 9 values of lambda.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 207 - 217.
2. Brownlee, K. A. (1965), ***Statistical Theory and Methodology in Science and Engineering***, Second Edition, John Wiley and Sons, Inc., pp. 282 - 285.
3. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 271 - 272.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the limits for lambda, sample size, and Type I error together with the test type. If NCASES is greater than one, record type 2 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5
2	DLAMBLL	Lower limit of lambda values (DLAMBLL ≥ 0.0)	1 - 8	F8.0
	DLINCR	Lambda value increment (DLINCR > 0.0)	9 - 16	F8.0
	DLAMBUL	Upper limit of lambda values (DLAMBUL ≥ 0.0)	17 - 24	F8.0

FNLL	Lower limit of sample size values (FNLL > 0)	25 - 32	F8.0
FNINCR	Sample size value increment (FNINCR > 0)	33 - 40	F8.0
FNUL	Upper limit of sample size values (FNUL > 0)	41 - 48	F8.0
ALPHALL	Lower limit of Type I error values (0.0 < ALPHALL < 1.0)	49 - 56	F8.0
AINCR	Type I error value increment (AINCR > 0.0)	57 - 64	F8.0
ALPHAUL	Upper limit of Type I error values (0.0 < ALPHAUL < 1.0)	65 - 72	F8.0
TYPE	Test type, i.e., =1, for two-sided test (hypothesis (1)) =2, for one-sided, right-tailed test (hypothesis (2)) =3, for one-sided, left-tailed test (hypothesis (3))	73 - 80	F8.0

One record type 2 is required for each case to be processed.

FVARPOW

PURPOSE

Program FVARPOW (Power of the F Test on Two Variances) evaluates the power function for tests of hypotheses regarding the equality of the variances of two normal populations. The tests are based on two independent random samples of sizes n_1 and n_2, respectively, from two normal populations having unknown variances σ_1^2 and σ_2^2. Let X_1 and X_2 be normal random variables with unknown means μ_1 and μ_2 and unknown variances σ_1^2 and σ_2^2. Then the sample means $\overline{X}_1$ and $\overline{X}_2$ are normal random variables with unknown means μ_1 and μ_2 and unknown variances σ_1^2/n_1 and σ_2^2/n_2, respectively, where $\overline{X}_{i.} = \sum_{j=1}^{n_i} X_{ij}/n_i$. These tests are commonly referred to as F tests for the equality of two variances (see References 1 and 2). The three relevant hypotheses and their corresponding test statistics, critical regions, and power functions are given below. In each case,

α = size of the critical region
= Prob(committing a Type I error)

and the test statistic

$$T_0 = F_0 = S_1^2/S_2^2$$

where S_i^2 is the unbiases estimator of the variance σ_i^2 and is given by

$$S_i^2 = \sum_{j=1}^{n_i} (X_{ij} - \overline{X}_i)^2/(n_i - 1) .$$

Define $\lambda = \sigma_1/\sigma_2$ and $c = 1/\lambda$. The probability density function of T_0 depends on the true value of the parameter λ ($\lambda = \lambda_0$ or $\lambda = \lambda_1 \neq \lambda_0$). $\lambda_0 = 1$ is the hypothesized value of λ. For $\lambda = \lambda_0$, this density will be denoted by $g(t_0 \mid \lambda_0)$. $g(t_0 \mid \lambda_0)$ is an F distribution with parameters ν_1 (numerator degrees of freedom) and ν_2 (denominator degrees of freedom) and is given by

$$g(t_0 \mid \lambda_0) = \frac{\Gamma[(\nu_1+\nu_2)/2](\nu_1/\nu_2)^{\nu_1/2}}{\Gamma(\nu_1/2)\Gamma(\nu_2/2)} \cdot \frac{t_0^{\nu_1/2-1}}{(1+\nu_1 t_0/\nu_2)^{(\nu_1+\nu_2)/2}} \quad , \quad t_0 > 0$$

Here $\nu_1 = n_1 - 1$ and $\nu_2 = n_2 - 1$. The critical region for the test is based on $g(t_0 \mid \lambda_0)$ and the power function is expressed in terms of λ_1 and $g(t_0 \mid \lambda_0)$.

* *

Hypothesis (1). H_0: $\sigma_1 = \sigma_2$

H_1: $\sigma_1 \neq \sigma_2$.

Critical Region: Reject H_0 if $T_0 < F_{\nu_1,\nu_2,\alpha/2}$

or if $T_0 > F_{\nu_1,\nu_2,1-\alpha/2}$ where $F_{\nu_1,\nu_2,1-\alpha/2}$ is the 100(1 - α/2)th percentage point of the F distribution with ν_1 and ν_2 degrees of freedom. Hence,

$$\alpha/2 = \int_{F_{\nu_1,\nu_2,1-\alpha/2}}^{\infty} g(t_0 \mid \lambda_0) dt_0 .$$

Power Function: $$\prod(\lambda_1, n_1, n_2) = \int_0^{c^2 F_{\nu_1,\nu_2,\alpha/2}} g(t_0 \mid \lambda_1) dt_0 + \int_{c^2 F_{\nu_1,\nu_2,1-\alpha/2}}^{\infty} g(t_0 \mid \lambda_0) dt_0 .$$

* *

Hypothesis (2): H_0: $\sigma_1 \leq \sigma_2$

H_1: $\sigma_1 > \sigma_2$

Critical region: Reject H_0 if $T_0 > F_{\nu_1,\nu_2,1-\alpha}$

Hence,

$$\alpha = \int_{F_{\nu_1,\nu_2,1-\alpha}}^{\infty} g(t_0 \mid \lambda_0) dt_0 .$$

Power Function: $$\prod(\lambda_1, n_1, n_2) = \int_{c^2 F_{\nu_1,\nu_2,1-\alpha}}^{\infty} g(t_0 \mid \lambda_0) dt_0 .$$

* *

Hypothesis (3): H_0: $\sigma_1 \geq \sigma_2$

H_1: $\sigma_1 < \sigma_2$

Critical Region: Reject H_0 if $T_0 < F_{\nu_1,\nu_2,\alpha}$

Hence,

$$\alpha = \int_0^{F_{\nu_1,\nu_2,\alpha}} g(t_0 \mid \lambda_0) dt_0 .$$

Power Function: $$\prod(\lambda_1, n_1, n_2) = \int_0^{c^2 F_{\nu_1,\nu_2,\alpha}} g(t_0 \mid \lambda_0) dt_0 .$$

FEATURES

FVARPOW allows the user to process up to 25 cases in a single computer run. Each case corresponds to a single range of the user-specified Type I errors. FVARPOW provides the user with the option of either computing a single power for a specified sample size pair or printing a table of power values. If the single power option is chosen, the user must also specify the type of test (i.e., one- or two-sided), a sample size pair, and a value of lambda, where lambda is the ratio of the true numerator population standard deviation to the true denominator population standard deviation. Hypotheses (2) and (3) are one-sided tests while hypothesis (1) is a two-sided test. If the power table option is chosen, the user instead specifies the test type, and a range of values for the numerator population sample size, Type I error, and lambda. In addition, a multiplicative factor must be provided which relates the denominator population sample size to the numerator population sample size. Under the power table option each page of FVARPOW output consists of the following:

* A power table for the specified test type and Type I error for up to 40 values of the pair (n_1, n_2) and up to 9 values of lambda.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 254 - 265.
2. Brownlee, K. A. (1965), ***Statistical Theory and Methodology in Science and Engineering***, Second Edition, John Wiley and Sons, Inc., pp. 285 - 288.
3. Freund, John E. (1962), ***Mathematical Statistics***, Prentice-Hall, Inc., pp. 273 - 274.

INPUT GUIDE

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the power table computation option. If the single power computation option is chosen, the third record type specifies the values of lambda, the numerator population sample size, the denominator population sample size, the Type I error, and the test type. If the power table option is chosen, the third record type is different and a fourth record type is also required. In this case, the third record type specifies the limits for lambda, the numerator population sample size, and Type I error together with the test type. The fourth record type specifies the value of the multiplicative factor relating the denominator population sample size to the numerator population sample size. If NCASES is greater than one, record types 2, 3 (3A or 3B), and, if necessary, 4 must be repeated for each case.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1 - 5	I5
2	IOPTION	Power computation option = 0, for power table output = 1, for single power value	1 - 5	I5
		The following record type is included only if IOPTION = 1.		
3A	DL(1)	Lambda value (DL(1) ≥ 0.0)	1 - 8	F8.0
	FN(1)	Numerator population sample size (FN(1) > 0)	9 - 16	F8.0
	FDEN	Denominator population sample size (FDEN > 0)	17 - 24	F8.0
	ALPH(1)	Type I error (0.0 < ALPH(1) < 1.00)	25 - 32	F8.0
	TYPE	Test type, i.e., =1, for one-sided test (hypothesis (2) or (3)) =2, for two-sided test (hypothesis (1))	33 - 40	F8.0

The following record type is included only if IOPTION = 0.

3B	DLAMBLL	Lower limit of lambda values (DLAMBLL ≥ 0.0)	1 - 8	F8.0
	DLINCR	Lambda value increment (DLINCR > 0.0)	9 - 16	F8.0
	DLAMBUL	Upper limit of delta values (DLAMBUL ≥ 0.0)	17 - 24	F8.0
	FNLL	Lower limit of numerator population sample size values (FNLL > 0)	25 - 32	F8.0
	FNINCR	Numerator population sample size value increment (FNINCR > 0)	33 - 40	F8.0
	FNUL	Upper limit of numerator population sample size values (FNUL > 0)	41 - 48	F8.0
	ALPHALL	Lower limit of Type I error values (0.0 < ALPHALL < 1.0)	49 - 56	F8.0
	AINCR	Type I error value increment (AINCR > 0.0)	57 - 64	F8.0
	ALPHAUL	Upper limit of Type I error values (0.0 < ALPHAUL < 1.0)	65 - 72	F8.0
	TYPE	Test type, i.e., =1, for two-sided test (hypothesis (1)) =2, for one-sided, right-tailed test (hypothesis (2)) =3, for one-sided, left-tailed test (hypothesis (3))	73 - 80	F8.0

Record type 4 is included only if IOPTION = 0.

4	FACTOR	Sample size multiplicative factor. Satisfies the relationship n_2 = FACTOR*n_1. (FACTOR > 0.0)	1 - 20	F8.0

One record type 2 and 3 (3A or 3B) is required for each case to be processed. One record type 4 is required for each case in which IOPTION = 0.

FEMPOW

PURPOSE

Program FEMPOW (Fixed Effects Model Power) computes the power of the test for a fixed effects model one-way analysis of variance (anova). The one-way anova fixed effects model is written

$$Y_{ij} = \mu + A_i + E_{ij}$$

with $i = 1, 2, ..., k$ (number of treatments)
and $j = 1, 2, ..., n_i$ (number of observations for treatment i).

The A_i's represent effects of k fixed or predetermined treatments and are subject to the constraint

$$\sum_{i=1}^{k} A_i = 0 .$$

The specific k treatment effects are the object of the analyst's interest for the fixed effects model. The E_{ij}'s represent random error components and are assumed normally and independently distributed with mean zero and variance σ^2. Although the one-way anova model accommodates different numbers of observations per treatment, the power computations in FEMPOW are computed only for the case where $n_i = n$, i.e., the number of observations per treatment is n. FEMPOW was formulated as a planning aid for the analyst who wishes to compare the effect of number of treatments and sample size per treatment upon power.

The null hypothesis of interest is that the treatment effect is equal to zero for each treatment. For the current model this is expressed as

$$H_0: \quad A_1 = A_2 = \cdots = A_k = 0$$

versus the alternative hypothesis

$$H_1: \quad \text{At least one } A_i \text{ is not equal to zero.}$$

Equivalent expressions can be obtained by defining the ith treatment mean as $\mu_i = \mu + A_i$. Then the null hypothesis becomes

$$H_0: \quad \mu_1 = \mu_2 = \cdots = \mu_k = 0$$

and the alternative hypothesis will be

$$H_1: \quad \text{At least two of the means are not equal.}$$

The test statistic is

$$F_0 = \frac{\text{Treatment Mean Square from Anova}}{\text{Error Mean Square from Anova}}$$

$$= \frac{n \sum_{i=1}^{k} (\overline{Y}_{i.} - \overline{Y}_{..})^2}{k-1} \Big/ \frac{\sum_{i=1}^{k} \sum_{j=1}^{n} (Y_{ij} - \overline{Y}_{i.})^2}{k(n-1)}$$

which follows an F distribution with $(k-1)$ and $k(n-1)$ degrees of freedom if H_0 is true. For a specified value of α, the probability of Type I error, one rejects H_0 if

$$F_0 > F(k-1, k(n-1), 1-\alpha) = 100(1-\alpha)\text{th percentile of the } F \text{ distribution with } k-1 \text{ and } k(n-1) \text{ degrees of freedom.}$$

In the fixed effects model one-way anova, power measures the ability of the test to detect differences in the treatment effects and is determined by calculating the probability of rejecting H_0 when H_1 is true. The power function is expressed as

$$\prod(\lambda, k, n) = \text{Prob}\left(F(\lambda, k-1, k(n-1)) > F(k-1, k(n-1), 1-\alpha) \mid \sum_{i=1}^{k} A_i^2 \neq 0 \right)$$

where $F(\lambda, k-1, k(n-1))$ represents the non-central F distribution with non-centrality parameter

$$\lambda = n \sum_{i=1}^{k} A_i^2 / (2\sigma^2)$$

and degrees of freedom, $k-1$ and $k(n-1)$. However, without the inconvenience of assigning specific values to each A_i, it is difficult to obtain meaningful power computations from the above formulation. Reference 1 presents expressions which allow upper and lower bounds of λ to be determined as a function of w, the range of treatment means. Using these bounds minimum and maximum values of power can then be computed for specified values of w, k, n, σ^2 and α. The analyst may easily evaluate the sensitivity of the test for the fixed effects model by exercising FEMPOW over appropriate sets of values for the given parameters. Additional discussion of the power function parameters is given in the COMMENTS section.

References 1 and 2 contain additional information about the one-way analysis of variance for the fixed effects model. Both references contain specific discussions as well as examples of the computation of power.

FEATURES

The output from FEMPOW is the power for the one-way fixed effects anova model. Power is be computed and printed as described below:

* A minimum and maximum value of power is computed for each specified combination of the power function parameters w, k, n, σ^2 and α.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 377 - 403.

2. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan Publishing Co., pp. 407 - 417 and pp. 463 - 468.

INPUT GUIDE

The specifications of the user-created input file are given below. The input file must contain all six record types.

Record Type	Variable	Description	Columns	Format
1	NM	Number of n values to be processed where n is the number of observations on each of the k treatments. (NM ≤ 10)	1-5	I5
	NK	Number of k values to be processed where k is the number of treatments being compared in the anova. (NK ≤ 10)	6-10	I5
	NW	Number of w values to be processed where w is the range of the means of the k treatment populations. (NW ≤ 7)	11-15	I5
	NSIG	Number of sigma (σ) values to be processed where sigma is the standard deviation of the random error component in the fixed effects anova model. (NSIG ≤ 7)	16-20	I5
	NALPHA	Number of alpha α values to be processed where alpha is the probability of a Type I error. (NALPHA ≤ 7)	21-25	I5

2	NS(I)	Enter NM (≤ 10) values of n, the number of observation per each of the k treatments.	1-50	10I5
3	K(I)	Enter NK (≤ 10) values of k, the number of treatments being compared in the anova.	1-50	10I5
4	W(I)	Enter NW (≤ 7) values of w, the range of the means of the treatment populations.	1-70	7F10.4
5	SIG(I)	Enter NSIG (≤ 7) values of σ, the standard deviation of the random error component in the fixed effects anova model. See COMMENTS section.	1-70	7F10.4
6	ALPHA(I)	Enter NALPHA (≤ 7) values of α, the probability of a Type I error.	1-70	7F10.4

COMMENTS

For cases where the power function parameters k, n, σ^2 and α are fixed, the value of power varies for each set of hypothesized treatment effects, A_i. Reference 1 provides expressions which yield minimum and maximum values of λ based upon the range of hypothesized treatment means, w, rather than the individual A_i's. These values of λ are

$$\lambda(\min) = nw^2/(4\sigma^2) \quad \text{and}$$

$$\lambda(\max) = \begin{cases} knw^2/(8\sigma^2), & \text{for k even} \\ (k^2-1)nw^2/(8k\sigma^2), & \text{for k odd.} \end{cases}$$

This allows the analyst a simpler way to present the alternative hypothesis, H_1, for power calculations. Once values of the other power function parameters are specified, only a value for the range of treatment means needs to be assigned rather than values for each mean.

The selection of values for the parameter σ, the standard deviation of the random error component, requires additional remarks. If the analyst has knowledge about the value of this parameter, then he can easily specify a realistic input value. In the absence of such information a value would have to be assumed. However, because λ is a function of the ratio of w to σ, it is possible to "standardize" w in units of σ. This can be done by using the value 1 for σ. For this approach the analyst needs to express w in terms of its ratio to σ rather than in absolute units.

REMPOW

PURPOSE

Program REMPOW (Random Effects Model Power) computes the power of the test for a random effects model one-way analysis of variance (anova). The one-way anova random effects model is written

$$Y_{ij} = \mu + A_i + E_{ij}$$

with $i = 1, 2, ..., k$ (number of treatments)
and $j = 1, 2, ..., n_i$ (number of observations for treatment i).

The A_i's represent effects of randomly chosen treatment levels which are assumed normally and independently distributed with mean zero and variance σ_A^2. The E_{ij}'s which represent random error components are also assumed normally and independently distributed with mean zero and variance σ^2. Although the one-way anova model accommodates different numbers of observations per treatment, the power computations in REMPOW are computed only for the case where $n_i = n$, i.e., the number of observations per treatment is n. REMPOW was formulated as a planning aid for the analyst who wishes to compare the effect of number of treatments and sample size per treatment upon power.

The null hypothesis of interest is that the variability for the population of treatment effects is zero. For the current model this is expressed as

$$H_0: \quad \sigma_A^2 = 0$$

versus the alternative hypothesis

$$H_1: \quad \sigma_A^2 > 0 .$$

The test statistic is

$$F_0 = \frac{\text{Treatment Mean Square from Anova}}{\text{Error Mean Square from Anova}}$$

$$= \frac{n \sum_{i=1}^{k} (\overline{Y}_{i.} - \overline{Y}_{..})^2}{k-1} \Big/ \frac{\sum_{i=1}^{k} \sum_{j=1}^{n} (Y_{ij} - \overline{Y}_{i.})^2}{k(n-1)}$$

which follows an F distribution with $(k-1)$ and $k(n-1)$ degrees of freedom if H_0 is true. For a specified value of α, the probability of Type I error, one rejects H_0 if

$F_0 > F(k-1, k(n-1), 1-\alpha) = 100(1-\alpha)$th percentile of the F distribution with $k-1$ and $k(n-1)$ degrees of freedom.

In the random effects model one-way anova, power measures the ability of the test to detect variation in the treatment effects and is determined by calculating the probability of rejecting H_0 when H_1 is true. The power function is expressed as

$$\Pi(\lambda, k, n) = \text{Prob}\left(F > \frac{F(k-1, k(n-1), 1-\alpha)}{\lambda^2} \mid \sigma_A^2 > 0 \right)$$

where $\lambda = (1 + n(\sigma_A^2 / \sigma^2))^{1/2}$. However, λ does not provide an easily interpretable concept with respect to the variance components. Two other parameters are defined for this program each of which possess this attribute. The first is $R = \sigma_A / \sigma$ which represents the ratio of the treatment standard deviation to the experimental (random) error standard deviation. The second parameter is $P = \left(\sqrt{\sigma^2 + \sigma_A^2} - \sigma\right) / \sigma$ which represents the proportionate increase in the total standard deviation of Y which can be attributed to variation between the treatments. At the users option λ and thus Π can be expressed in terms of either of the parameters, R or P. Both R and P provide convenient measures for relating σ_A to σ which assists the analyst in evaluating the sensitivity of the test for a given scenario. The analyst specifies values of R and/or P of interest and an acceptable value of α. He then chooses a set of values for the sample size n and for the number of treatments k which are appropriate for the available experimental resources. The resulting power computations enable him to relate sample size to power for the specified values of R and/or P, k and α. The concepts of R, P and λ are discussed in more detail in the COMMENTS section.

References 1 and 2 contain additional information about the one-way analysis of variance for the random effects model. Both references contain specific discussions as well as examples of the computation of power.

FEATURES

The output from REMPOW is the power for the one-way random effects anova model. Power can be computed and printed as described below:

* Power can be computed and printed as a function of $R = \sigma_A / \sigma$.
* Power can be computed and printed as a function of $P = \left(\sqrt{\sigma^2 + \sigma_A^2} - \sigma\right) / \sigma$.
* Power can be computed and printed as a function of R and/or P on the same computer run.
* For a specified value of α (probability of Type I error), the power can be computed and printed for up to 9 values of the number of treatments k, up to 40 values of the sample size n and up to 40 values of either R and/or P on the same computer run.

REFERENCES

1. Bowker, A. H. and Lieberman, G. J. (1972), ***Engineering Statistics***, Second Edition, Prentice-Hall, Inc., pp. 377 - 403.

2. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan Publishing Co., pp. 454 - 460 and pp. 463 - 468.

INPUT GUIDE

The specifications of the user-created input file are given below. The input file must contain all five record types. Multiple cases are handled by repeating record types 2 through 5 for each case.

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases to be processed. A case represents a set of record types 2 through 5. For a single case the program computes power as a function of *R* or *P*.	1-5	I5
2	NK	Number of k values to be processed where k is the number of treatments being compared in the anova. (NK ≤ 9)	1-5	I5
	NN	Number of n values to be processed for this case where n is the number of observations on each of the k treatments. (NN ≤ 40)	6-10	I5
	NRAT	Number of *R* values to be processed for this case. If NRAT > 0, then NPCT (columns 16-20) must be 0. (NRAT ≤ 40)	11-15	I5
	NPCT	Number of *P* values to be processed for this case. If NPCT > 0, then NRAT (columns 11-15) must be 0. (NPCT ≤ 40)	16-20	I5

	ICASE	=1, Power will be computed as a function of R. =2, Power will be computed as a function of P.	21-25	I5
	ALPHA	Enter the value of α, the probability of making a Type I error. A Type I error is made by wrongly rejecting $H_0: \sigma_A^2 = 0$.	26-35	F10.5
3	FK(I) I=1,NK	Enter NK (≤ 9) values of k, the number of treatments being compared in the anova.	1-80	16F5.0
4	FN(I) I=1,NN	Enter NN (≤ 40) values of n, the number of observations on each of the k treatments.	1-80	16F5.0
5	RATIO(I) I=1,NRAT	If ICASE = 1 (columns 21-25, record type 2), enter NRAT (≤ 40) values of R.	1-80	16F5.0
	or PCT(I) I=1,NPCT	If ICASE = 2, enter NPCT (≤ 40) values of P. P is entered as a proportion.		

COMMENTS

The power function Π given earlier contains the parameter

$$\lambda = (1 + n(\sigma_A^2 / \sigma^2))^{1/2}.$$

However, in order to facilitate the use of the power computations the power function in the program is formulated in terms of either one of two other parameters. They are

(1) $R = \sigma_A / \sigma$ and

(2) $P = \left(\sqrt{\sigma^2 + \sigma_A^2} - \sigma\right) / \sigma$

If the user chooses to compute power as a function of R, this simply means that σ_A^2 / σ^2 is replaced by R in the power function expression. In the second case it can be shown that

$$\sigma_A^2 / \sigma^2 = (1 + P)^2 - 1,$$

and this becomes the necessary substitution within the program.

PROBABILITY

EVALUATION

BINVARP

PURPOSE

Program BINVARP computes binomial probabilities for the case in which the probability of success on a single trial is allowed to vary from trial to trial. Specifically, BINVARP is designed to compute the probability of obtaining x successes in a sequence of *NLIM* trials where the success probabilities at successive trials are $p_1, p_2, \ldots, p_{nlim}$, respectively. The classical situation in which trials are independent and the probability of success remains constant from trial to trial is referred to as "Bernoulli trials" in the literature. The situation addressed in BINVARP is sometimes referred to as "Bernoulli trials with variable probabilities", but it shows up more frequently in the literature under the confusing name of "Poisson trials" (see Reference 1).

As an example of a situation to which BINVARP is applicable, consider the following scenario. Suppose a weapon system attempts to destroy an incoming missile before the missile damages the weapon system. A sequence of *NLIM* rounds is fired at the missile, each having a different probability of hit due to the changing range of the missile. If K hits are required to destroy the missile, what is the probability that the missile is destroyed prior to impact? BINVARP can be utilized to answer this question. Note that in this example the probability of hit is range dependent. The maximum achievable range is usually divided into subintervals, and a constant probability of hit is associated with each. BINVARP assumes that the number of subintervals, and, hence, the number of distinct trial probabilities, is less than or equal to the total number of rounds fired.

BINVARP employs a recursive computing algorithm which is based on the concept of a probability generating function. For a detailed development of this algorithm see References 2 and 3.

FEATURES

BINVARP allows the user to process multiple cases in a single computer run. For each case BINVARP outputs the following information:

* The total number of binomial trials (*NLIM*), the number of distinct trial probabilities, and a frequency table listing the value of each distinct probability and the number of trials in which it is used.

* A table of probabilities displaying the number of successes, (K), the probability of obtaining exactly K successes, and the cumulative probability of obtaining K or less successes. The user can control the size of this table by specifying the integer index (KLAST) of the last probability he wishes to have printed out. Hence, the resultant table will always include probabilities for 0 through KLAST successes.
* The mean and standard deviation of the number of successes in *NLIM* binomial trials.

REFERENCES

1. Feller, William (1965), ***An Introduction to Probability Theory and Its Applications***, Volume I, Second Edition, John Wiley & Sons, Inc., pp. 205 - 206 and pp. 216 - 217.
2. Thomas, M. A. and Taub, A. E. (1975), ***Binomial Trials With Variable Probabilities***, NSWC/DL TN-DK-25/75, NSWC, Dahlgren, VIRGINIA 22448.
3. Thomas, M. A. and Taub, A. E. (1982), ***Calculating Binomial Probabilities When the Trial Probabilities are Unequal***, Journal of Statistical Computation and Simulation, Volume 14, pp. 125 - 131.

INPUT GUIDE

Some changes have been made to the original program since the date of Reference 2. For this reason the input guide specified below should take precedence over the one given in Reference 2.

The user must create a data file containing the following record types. The first record type specifies the number of cases (NCASES) to be run. The second record type specifies the number of distinct trial probabilities, the index of the last probability to be printed out, and a flag value indicating whether or not the number of distinct trial probabilities equals the total number of trials (*NLIM*). The third record type is included only if the number of distinct trial probabilities is less than *NLIM*. It contains the frequencies of occurrence of each of the distinct trial probabilities. Record type four is used to input the values of the distinct trial probabilities. If NCASES is greater than one, record types 2 through 4 must be repeated for each case. The value of *NLIM* is determined within BINVARP and, hence, is not to be input by the user. Nevertheless, BINVARP is limited to a maximum of 325 trials, so the user must ensure that $NLIM \leq 325$ for each case he runs. For $NLIM > 325$ the user is referred to Reference 1 for an approximation.

More specifically the required data file is constructed as follows:

Record Type	Variable	Description	Columns	Format
1	NCASES	Number of cases	1-5	I5
2	NP	Number of distinct trial probabilities	1-5	I5
	KLAST	Index of the last probability to be printed out, i.e., the probability of obtaining KLAST successes in NLIM trials.	6-10	I5
	NSS	=0, NP < NLIM =1, NP = NLIM	11-15	I5
		Record type 3 is included only if NP < NLIM.		
3	N(1)	Frequency of occurrence of 1st distinct trial probability	1-5	I5
	N(2)	Frequency of occurrence of 2nd distinct trial probability	6-10	I5
	....	.....	.....	...
	N(16)	Frequency of occurrence of 16th distinct trial probability	76-80	I5
	Additional records are required in the event that there are more than 16 distinct trial probabilities.			
4	P(1)	1st distinct trial probability	1-10	F10.0
	P(2)	2nd distinct trial probability	11-20	F10.0
	....	.....	.....	...
	P(8)	8th distinct trial probability	71-80	F10.0
	Additional records are required in the event that there are more than 8 distinct trial probabilities.			

COMMENTS

The user should recognize that BINVARP assumes that the NP distinct trial probabilities have been ordered in some meaningful way. The program's algorithm employs the first distinct trial probability for the first N(1) trials, the second distinct trial probability for the next N(2) trials, and so forth until the total number of trials has been exhausted.

NEGBIN

PURPOSE

Program NEGBIN (Negative Binomial) evaluates the negative binomial probability distribution and cumulative distribution functions for user specified parameter values. The form chosen for the negative binomial distribution is

$$P(X=n)=\binom{n-1}{r-1}p^r(1-p)^{n-r} \quad , \quad n=r,r+1,\ldots$$

where p is the probability of success on a single trial and X is number of the trial on which the rth success occurs. The corresponding cumulative distribution function is given by

$$F(n)=P(X\le n)=\sum_{t=r}^{n}\binom{t-1}{r-1}p^r(1-p)^{t-r}.$$

The parameters specified by the user are p and r. A value for $F(n)$, e.g., $F(n) = C$, must also be specified, and the program will cumulate probabilities until this value is obtained. With the corresponding value of n determined one can state that the Prob(rth success occurs on or before the nth trial) is C.

This procedure is oftentimes used in round requirement studies where one has probability p of hitting a target with a single round, and r hits are required for a target kill. One can specify a required kill probability C and ascertain the number of rounds n to achieve it.

FEATURES

NEGBIN features include the following:

* $P(n)$ and $F(n)$ evaluations for each $r \le n \le N$ where N is the smallest integer such that $F(n) \ge C$.
* Provisions for up to 1000 multiple runs.

REFERENCES

1. Feller, William (1957). ***An Introduction to Probability Theory and Its Applications***, John Wiley and Sons, pp. 155-156.
2. Johnson, Norman L. and Kotz, Samuel (1969). ***Distributions in Statistics - Discrete Distributions***, Houghton Mifflin Company, pp. 122-142.

INPUT GUIDE

The user-created input file consists of two record types as described below.

Record Type	Variable	Description	Columns	Format
1	K	Number of cases. (K < 1000)	1-3	I3
2	P	Single trial probability of success.	1-10	F10.5
	KR	Negative binomial parameter *r*.	16-20	I5
	CONF	Specification for $F(n)$, i.e., the probability or confidence C which terminates the negative binomial cumulation.	25-35	F10.5

Record 2 is repeated for each case.

CONFIDENCE

LIMIT

EVALUATION

BINCL

PURPOSE

Program BINCL (Binomial Confidence Limits) evaluates confidence limits and confidence bounds (one-sided confidence limits) for the binomial parameter P. This parameter denotes the true probability of success (failure) in a binomial experiment, i.e., an experiment consisting of N independent and identical trials where each trial results in a success or failure. The bounds and limits are computed as a function of the number of trials, N, and the number of successes (failures), R, in the experiment. The equations defining the upper, $\overline{P}$, and lower, $\underline{P}$, $100\gamma\%$ confidence bounds for P are given by (Reference 2):

$$\sum_{x=0}^{R}\binom{N}{x}\overline{P}^{x}(1-\overline{P})^{N-x}=1-\gamma \qquad (upper)$$

$$\sum_{x=R}^{N}\binom{N}{x}\underline{P}^{x}(1-\underline{P})^{N-x}=1-\gamma \qquad (lower).$$

In order to find the two-sided $100\gamma\%$ confidence limits, γ is replaced with $(1-\gamma)/2$ in the above equations. These equations are solved using the relationships between the cumulative binomial distribution and the incomplete beta function ratio. The relationships are given by

$$\sum_{x=0}^{R}\binom{N}{x}\overline{P}^{x}(1-\overline{P})^{N-x}=1-I_{\overline{P}}(R+1,N-R)$$

$$\sum_{x=R}^{N}\binom{N}{x}\underline{P}^{x}(1-\underline{P})^{N-x}=1-I_{\underline{P}}(R,N-R+1)$$

where

$$I_x(a,b)=\int_0^x t^{a-1}(1-t)^{b-1}/\left(\frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}\right)dt.$$

The $100\gamma\%$ upper confidence bound on P is found, therefore, by solving

$$I_{\overline{P}}(R+1,N-R)=\gamma$$

for $\overline{P}$. Similarly, the $100\gamma\%$ lower confidence bound is found by solving

$$I_{\underline{P}}(R,N-R+1)=\gamma$$

for $\underline{P}$. To find the $100\gamma\%$ confidence limits on P, the above equations are solved for $\overline{P}$ and $\underline{P}$ with $(1+\gamma)/2$ substituted for γ on their right hand sides. The solution of $I_x(a,b)=\gamma$ for x in the above is accomplished with an inverse incomplete beta function routine which utilizes the routine developed by DiDonato and Jarnagin (Reference 1) to evaluate $I_x(a,b)$.

FEATURES

The confidence bounds and limits are computed, tabled, and plotted according to any one of three options.

* Option 1 - For all values of $R = 0$ to N in steps of J units.
* Option 2 - For all values of $R = R$ to N in steps of J units.
* Option 3 - For all values of $R = 0$ to R in steps of J units.

The plots show the confidence bounds and limits as a function of the number of successes (failures) R. The table of printed values is formatted to accommodate a maximum of 51 values of R. However, the plots will accommodate an unrestricted number of values. Multiple values of γ and N may be input, and the program will compute a table and a plot for each combination of values.

REFERENCES

1. DiDonato, A. R. and Jarnagin, M. P., Jr. (1966), ***A Method for Computing the Incomplete Beta Function Ratio***, NWL Report No. 1949, Revised, Naval Surface Warfare Center, Dahlgren, VA
2. Mood, Alexander M. and Graybill, Franklin A. (1963), ***Introduction to the Theory of Statistics***, Second Edition, McGraw-Hill Book Company, Inc., pp. 260-262.

INPUT GUIDE

The user-created input file consists of three record types as described below.

Record Type	Variable	Description	Columns	Format
1	NG	Number of values of γ to be input. (NG ≤ 10)	1-10	I10
	NNS	Number of values of N to be input. (NNS ≤ 10)	11-20	I10

2	GAMMA(I)	NG γ values, GAMMA(I), I = 1, NG. If NG > 8, a second record type 2 will be required.	1-80	8E10.4
3	NN(1)	First of NNS values of *N*.	1-5	I5
	IOP(1)	Option number associated with the first value of *N*, IOP(1) = 1, 2, or 3. (See FEATURES section.)	6-10	I5
	JAY(1)	Step size associated with the first value of *N*.	11-15	I5
	IR(1)	Value of *R* for the first value of *N* when option 2 or 3 is specified.	16-20	I5

If NNS > 1, an additional NNS - 1 type 3 records will be required.

CEPCL

PURPOSE

Program CEPCL (Circular Probable Error Confidence Limits) computes point and interval estimates of the 100Pth circular percentile for a bivariate normal probability density function. (Program SEPCL, described elsewhere in this report, computes estimates of the 100Pth spherical percentile for the trivariate normal probability density function.) The program was designed for the analysis of fall-of-shot data but is applicable to any data which is drawn from the bivariate normal probability density function

$$f(x_1, x_2) = \frac{|\mathbf{V}|^{-1/2}}{2\pi} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})'\mathbf{V}^{-1}(\mathbf{x}-\boldsymbol{\mu})} \quad , -\infty < x_i < \infty .$$

In this expression, the mean of the distribution is at $\boldsymbol{\mu} = (\mu_1, \mu_2)$, and the covariances of the x_i are represented by the 2×2 variance-covariance matrix **V**. The matrix **V** is in standard form with the ith diagonal element representing the variance of x_i and the (i,j)th off-diagonal element representing the covariance between x_i and x_j.

By definition, the CEP is the radius of the 50 percent circle (the radius of the circle which contains .50 probability) centered on the target center. In this program, CEP is defined as an origin-centered circle which implies that the target center is located at the Cartesian origin. The program does not restrict one to the 50 percent circle but allows the percentage 100P to be specified by the user. The radius of this origin-centered circle which contains 100P percent of the bivariate probability is designated RP. Hence, for P = .50, RP coincides with the classical CEP.

CEPCL has two user-controlled modes of operation, the parameter mode and the estimation mode. In the parameter mode, the user has complete knowledge of the values of $\boldsymbol{\mu}$ and **V** and is interested only in obtaining the solution for RP, the 100Pth circular percentile. This is a numerical integration problem for which the software has been formulated in HP BASIC by DiDonato (Reference 3). For use in STATLIB, this software has been converted to FORTRAN 77 by Johnson (Reference 5). Hence, given the values of $\boldsymbol{\mu}$ and **V**, one can obtain an exact numerical solution for RP. Confidence limits on RP are not required since one is essentially 100% confident that the solution is the 100Pth circular percentile for the bivariate normal density in question. In the estimation mode, one's information regarding the parameters is in the form of n observed pairs (x_1, x_2). In the context of weapon accuracy analysis, each observed pair would represent an impact point in two space assuming target center is at (0, 0). These data are used to estimate the parameters $\boldsymbol{\mu}$ and **V** which are subsequently used as input to obtain a point estimate of RP. This is achieved using the numerical integration procedure referenced above. To obtain an interval estimate of RP, i.e., to obtain confidence limits for RP, one needs an analytical form for

this estimate of RP. This is provided with an RP approximation formulated by Grubbs (Reference 4). Using this form, $100\gamma\%$ confidence limits on RP are provided by an approximation formulated by Taub and Thomas (Reference 6).

In the parameter case, both the exact numerical integration solution and the analytical approximation to RP are provided. In the estimation case, both solutions are also provided as point estimates as well as the confidence limits for RP. While the approximate solution is not really necessary unless confidence limits are desired, it is included to provide a relatively simple analytical form for RP and to show its closeness to the exact solution. The approximation to RP and the confidence limit evaluation are discussed in the COMMENTS section.

To apply the approximation (as well as the exact numerical evaluation), it is assumed that **V** is diagonal, i.e., that the x_i are statistically independent. If **V** is not diagonal on input (or if **V** is estimated from observed data), independence is induced by rotating **V** by means of an orthogonal transformation. This rotation will be discussed in the COMMENTS section. At this point, it is sufficient to know that the evaluation of RP and its confidence limits are based on values of the means μ_1, μ_2 and variances σ_1^2, σ_2^2, either input values (**V** diagonal), transformed values (**V** not diagonal) or estimated values which may or may not have been transformed.

FEATURES

CEPCL features include the two modes of operation discussed above. In the parameter mode, the user specifies whether **V** is or is not diagonal. If diagonal, only the diagonal elements of **V**, i.e., the variances are input. Otherwise, the entire 2×2 matrix is input. In the estimation mode, input consists of n observed pairs from which μ and **V** are estimated. However, there are no internal tests to ascertain if $\mu = 0$, if the off-diagonal elements of **V** are zero, and if zero whether the variances are equal. The user controls how the estimates of the parameters are incorporated in subsequent evaluations by input assumptions which should be based on external testing. Therefore, in the estimation mode, it is recommended that two runs be made. The first should be made without any simplifying assumptions. The output from this run can then be used to perform significance tests on the parameters μ and **V**. A subsequent run should be made if these test results permit any simplifying assumptions.

Other features include the following:

* A listing of the first 50 data points (estimation mode).
* A printout of the input mean vector μ and covariance matrix **V** (parameter mode).

* A printout of the mean vector μ and covariance matrix **V** after rotation (**V** not diagonal).
* A printout of the mean vector μ and covariance matrix **V** under user specified assumptions (estimation mode).
* A printout of input specifications including user specified assumption (estimation mode).
* Computation and printout of RP (parameter mode) or $\hat{RP}$ (estimation mode), confidence limits on RP (estimation mode), and degrees of freedom for the chi-square approximations.

REFERENCES

1. Anderson, T. W. (1958), ***An Introduction to Multivariate Statistical Analysis***, John Wiley and Sons, Inc., p. 19.
2. Browne, E. T. (1958), ***Introduction to the Theory of Determinants and Matrices***, The University of North Carolina Press, p. 88 and p. 106.
3. DiDonato, A. R. (1987), ***Integration of the Trivariate Normal Distribution Over an Offset Sphere and an Inverse Problem***, NSWC TR 87-27, NSWC, Dahlgren, VIRGINIA 22448
4. Grubbs, F. E. (1964), "Approximate Circular and Non-Circular Offset Probabilities of Hitting," Operations Research, Vol. 12, No. 1.
5. Johnson, G. M. (1989), ***FORTRAN Conversion of Trivariate Normal Integration Over an Offset Sphere***, unpublished NSWC TN, NSWC, Dahlgren, VIRGINIA 22448
6. Taub, A. E. and Thomas, M. A. (1983), ***Confidence Intervals for CEP When the Errors are Elliptical Normal***, NSWC TR 83-205, NSWC, Dahlgren, VIRGINIA 22448

INPUT GUIDE

The user-created input file consists of four record types in either mode of operation.

Record Type	Variable	Description	Columns	Format
1	L1	=0, parameter mode =1, estimation mode.	1-2	I2

Use records 2, 3, and 4 only if L1 = 1.

2	FORMD	Data input format (in parentheses).	1-80	8A10
3	N	Number of observed data points.	1-5	I5
		($N \leq 10{,}000$)		
	IDIAG	=0, analysis assumes V diagonal.	6-10	I5
		=1, analysis assumes V not diagonal.		
	IMU	=0, analysis assumes mu is zero.	11-15	I5
		=1, analysis assumes mu is not zero.		
	IEQ	=0, analysis assumes variances equal.	16-20	I5
		=1, analysis assumes variances not equal.		
	P	Circular proportion requested.	21-30	F10.8
		($0 < P < 1$)		
	CCOEF	Confidence coefficient for confidence limits on RP.	31-35	F5.4
4		Data according to format FORMD.		
		Use records 5, 6, and 7 or 8 only if L1 = 0.		
5	IDIAG	=0, V is diagonal.	1-5	I5
		=1, V is not diagonal.		
	P	Circular proportion requested.	6-15	F10.8
		($0 < P < 1$)		
6	MU	Mean vector elements MU(I), I = 1, 2.	1-40	2F20.10
		Use record 7 if IDIAG = 0.		
7	V	Diagonal elements of V, V(I,I), I = 1, 2.	1-40	2F20.10
		Use record 8 if IDIAG = 1.		
8	V	Elements of V, one row per line.		
		V(1,J), J = 1, 2	1-40	2F20.10
		V(2,J), J = 1, 2	1-40	2F20.10

COMMENTS

A problem in CEPCL (estimation mode) is that of approximating *RP* with an analytical form which can subsequently be used to obtain confidence limits on *RP*. The problem can be stated as finding *RP* such that

$$\text{Prob}\left(\sum_i x_i^2 < (RP)^2\right) = P$$

under the bivariate normal assumptions on the x_i. To solve this problem analytically, we need the distribution of $\Psi^2 = \sum_i x_i^2$. It turns out that Ψ^2 is a weighted sum of non-central chi-square variables with a probability distribution which is too complicated to be useful. However, this distribution converges to a chi-square distribution as the σ_i^2 approach equality and the μ_i approach zero. Hence, the distribution of $\nu\Psi^2/E(\Psi^2)$ is approximated with a chi-square distribution with ν degrees of freedom. The degrees of freedom are obtained by the "method of matching moments," i.e., ν is the solution of the equation which equates the variance of $\nu\Psi^2/E(\Psi^2)$ with the variance of a chi-square with ν degrees of freedom. The solution to this equation and the results of the approximation are shown below:

$$(RP)^2 = K\left(\sum_i \sigma_i^2 + \sum_i \mu_i^2\right)/2$$

where

$$K = 2\chi^2_{\nu,P}/\nu\,.$$

In this expression, $\chi^2_{\nu,P}$ is the 100Pth percentile of a chi-square distribution with ν degrees of freedom where

$$\nu = 2m^2/c$$

and where *m* and *c* are the functions of the μ_i and σ_i^2 below:

$$m = 1 + \sum_i \mu_i^2 / \left(\sum_i \sigma_i^2\right)$$

$$c = 2\left(\sum_i \sigma_i^4 + 2\sum_i \sigma_i^2\mu_i^2\right) / \left(\sum_i \sigma_i^2\right)^2 .$$

The user is referred to Reference 4 for the complete derivation.

In the estimation mode, the value *RP* is actually an estimate of the true *RP*. Using the circumflex (or hat) convention to denote estimates of parameters, one would write

$$(R\hat{P})^2 = K\left(\sum_i \hat{\sigma}_i^2 + \sum_i \hat{\mu}_i^2\right)/2\,.$$

To place confidence limits on *RP*, the distribution of

$$\nu_2(\hat{RP})^2/(RP)^2$$

is approximated with a chi-square distribution with ν_2 degrees of freedom using the "method of matching moments" described above. It is found that ν_2 has value

$$\nu_2 = 2n^2m^2/\left(nc + 2\sum_i \sigma_i^4/\left(\sum_i \sigma_i^2\right)^2\right).$$

The approximate $100\gamma\%$ confidence limits for RP are then taken as

$$\left(\frac{\hat{RP}}{\left[\chi^2_{\nu_2,\frac{1+\gamma}{2}}/\nu_2\right]^{1/2}}, \frac{\hat{RP}}{\left[\chi^2_{\nu_2,\frac{1-\gamma}{2}}/\nu_2\right]^{1/2}}\right).$$

The user is referred to Reference 6 for the complete derivation of RP confidence limits and an evaluation of their accuracy.

A brief discussion regarding the rotation of non-diagonal covariance matrices will aid the user in understanding his output. CEPCL is applicable to any problem where the errors are bivariate normal whether **V** is or is not diagonal. However, the routines for computing RP (exact or approximate) require that **V** be diagonal, i.e., that the x_i be independent random variables. Hence, if **V** is not diagonal on input (parameter input or an estimate from input data), a rotation must be performed which removes the off-diagonal elements from **V** and adjusts the diagonal elements accordingly. This process is based on the following principles: If the matrix **V** is symmetric there exists an orthogonal matrix **W** such that **W′VW** = **D** where **D** is diagonal (Reference 2). (An orthogonal matrix **W** is square and defined such that **WW′** = **I**.) Also, if the random vector **x** has a multivariate normal distribution with mean μ and covariance matrix **V**, i.e., if

$$\mathbf{x} \sim N(\mu, \mathbf{V}),$$

and if **y** = **W′x**, then it follows from normal theory (Reference 1) that

$$\mathbf{y} \sim N(\mathbf{W}'\mu, \mathbf{W}'\mathbf{V}\mathbf{W}).$$

Hence, if **x** is multivariate normal, it follows that there exists an orthogonal **W** such that **y** = **W′x** is multivariate normal with mean **W′**μ and diagonal covariance matrix **D** = **W′VW**. Furthermore, since **W** is orthogonal, $\sum_i y_i^2 = \sum_i x_i^2$ so that

$$\text{Prob}\left(\sum_i x_i^2 < (RP)^2\right) = \text{Prob}\left(\sum_i y_i^2 < (RP)^2\right).$$

The result is that given **W**, one can transform from **x** to **y** (thus obtaining a new mean and a new diagonal covariance matrix), and the integral over any circular region is not affected by the transformation. The orthogonal matrix **W** is constructed in CEPCL by finding the eigenvalues of **V** and the eigenvector associated with each eigenvalue, and then taking **W** as the matrix of eigenvectors.

SEPCL

PURPOSE

Program SEPCL (Spherical Probable Error Confidence Limits) computes point and interval estimates of the 100Pth spherical percentile for a trivariate normal probability density function. (Program CEPCL, described elsewhere in this report, computes estimates of the 100Pth circular percentile for the bivariate normal probability density function.) The program was designed for the analysis of fall-of-shot data but is applicable to any data which is drawn from the trivariate normal probability density function

$$f(x_1,x_2,x_3)=\frac{|\mathbf{V}|^{-1/2}}{(2\pi)^{3/2}}e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})\mathbf{V}^{-1}(\mathbf{x}-\boldsymbol{\mu})} \quad ,-\infty<x_i<\infty .$$

In this expression, the mean of the distribution is at $\boldsymbol{\mu}=(\mu_1,\mu_2,\mu_3)$, and the covariances of the x_i are represented by the 3×3 variance-covariance matrix **V**. The matrix **V** is in standard form with the ith diagonal element representing the variance of x_i and the (i,j)th off-diagonal element representing the covariance between x_i and x_j.

By definition, the SEP is the radius of the 50 percent sphere (the radius of the sphere which contains .50 probability) centered on the target center. In this program, SEP is defined as an origin-centered sphere which implies that the target center is located at the Cartesian origin. The program does not restrict one to the 50 percent sphere but allows the percentage 100P to be specified by the user. The radius of this origin-centered sphere which contains 100P percent of the trivariate probability is designated *RP*. Hence, for P = .50, *RP* coincides with the classical SEP.

SEPCL has two user-controlled modes of operation, the parameter mode and the estimation mode. In the parameter mode, the user has complete knowledge of the values of $\boldsymbol{\mu}$ and **V** and is interested only in obtaining the solution for *RP*, the 100Pth spherical percentile. This is a numerical integration problem for which the software has been formulated in HP BASIC by DiDonato (Reference 3). For use in STATLIB, this software has been converted to FORTRAN 77 by Johnson (Reference 5). Hence, given the values of $\boldsymbol{\mu}$ and **V**, one can obtain an exact numerical solution for *RP*. Confidence limits on *RP* are not required since one is essentially 100% confident that the solution is the 100Pth spherical percentile for the trivariate normal density in question. In the estimation mode, one's information regarding the parameters is in the form of n observed 3-tuples (x_1,x_2,x_3). In the context of weapon accuracy analysis, each observed 3-tuple would represent an impact point in three space assuming target center is at (0, 0, 0). These data are used to estimate the parameters $\boldsymbol{\mu}$ and **V** which are subsequently used as input to obtain a point estimate of *RP*. This is achieved using the numerical integration procedure referenced above. To

obtain an interval estimate of *RP*, i.e., to obtain confidence limits for *RP*, one needs an analytical form for this estimate of *RP*. This is provided with an *RP* approximation formulated by Grubbs (Reference 4). Using this form, $100\gamma\%$ confidence limits on *RP* are provided by an approximation formulated by Taub and Thomas (Reference 6).

In the parameter case, both the exact numerical integration solution and the analytical approximation to *RP* are provided. In the estimation case, both solutions are also provided as point estimates as well as the confidence limits for *RP*. While the approximate solution is not really necessary unless confidence limits are desired, it is included to provide a relatively simple analytical form for *RP* and to show its closeness to the exact solution. The approximation to *RP* and the confidence limit evaluation are discussed in the COMMENTS section.

To apply the approximation (as well as the exact numerical evaluation), it is assumed that **V** is diagonal, i.e., that the x_i are statistically independent. If **V** is not diagonal on input (or if **V** is estimated from observed data), independence is induced by rotating **V** by means of an orthogonal transformation. This rotation will be discussed in the COMMENTS section. At this point, it is sufficient to know that the evaluation of *RP* and its confidence limits are based on values of the means (μ_1, μ_2, μ_3). and variances $(\sigma_1^2, \sigma_2^2, \sigma_3^2)$. either input values (**V** diagonal), transformed values (**V** not diagonal) or estimated values which may or may not have been transformed.

FEATURES

SEPCL features include the two modes of operation discussed above. In the parameter mode, the user specifies whether **V** is or is not diagonal. If diagonal, only the diagonal elements of **V**, i.e., the variances are input. Otherwise, the entire 3×3 matrix is input. In the estimation mode, input consists of n observed 3-tuples from which $\boldsymbol{\mu}$ and **V** are estimated. However, there are no internal tests to ascertain if $\boldsymbol{\mu} = 0$, if the off-diagonal elements of **V** are zero, and if zero whether the variances are equal. The user controls how the estimates of the parameters are incorporated in subsequent evaluations by input assumptions which should be based on external testing. Therefore, in the estimation mode, it is recommended that two runs be made. The first should be made without any simplifying assumptions. The output from this run can then be used to perform significance tests on the parameters $\boldsymbol{\mu}$ and **V**. A subsequent run should be made if these test results permit any simplifying assumptions.

Other features include the following:

* A listing of the first 50 data points (estimation mode).
* A printout of the input mean vector $\boldsymbol{\mu}$ and covariance matrix **V** (parameter mode).

* A printout of the mean vector μ and covariance matrix **V** after rotation (**V** not diagonal).
* A printout of the mean vector μ and covariance matrix **V** under user specified assumptions (estimation mode).
* A printout of input specifications including user specified assumption (estimation mode).
* Computation and printout of *RP* (parameter mode) or $\hat{RP}$ (estimation mode), confidence limits on *RP* (estimation mode), and degrees of freedom for the chi-square approximations.

REFERENCES

1. Anderson, T. W. (1958), ***An Introduction to Multivariate Statistical Analysis***, John Wiley and Sons, Inc., p. 19.
2. Browne, E. T. (1958), ***Introduction to the Theory of Determinants and Matrices***, The University of North Carolina Press, p. 88 and p. 106.
3. DiDonato, A. R. (1987), ***Integration of the Trivariate Normal Distribution Over an Offset Sphere and an Inverse Problem***, NSWC TR 87-27, NSWC, Dahlgren, VIRGINIA 22448
4. Grubbs, F. E. (1964), "Approximate Circular and Non-Circular Offset Probabilities of Hitting," Operations Research, Vol. 12, No. 1.
5. Johnson, G. M. (1989), ***FORTRAN Conversion of Trivariate Normal Integration Over an Offset Sphere***, unpublished NSWC TN, NSWC, Dahlgren, VIRGINIA 22448
6. Taub, A. E. and Thomas, M. A. (1983), ***Confidence Intervals for CEP When the Errors are Elliptical Normal***, NSWC TR 83-205, NSWC, Dahlgren, VIRGINIA 22448

INPUT GUIDE

The user-created input file consists of four record types in either mode of operation.

Record Type	Variable	Description	Columns	Format
1	L1	=0, parameter mode =1, estimation mode.	1-2	I2

Use records 2, 3, and 4 only if L1 = 1.

2	FORMD	Data input format (in parentheses).	1-80	8A10
3	N	Number of observed data points.	1-5	I5
		($N \leq 10{,}000$)		
	IDIAG	=0, analysis assumes V diagonal.	6-10	I5
		=1, analysis assumes V not diagonal.		
	IMU	=0, analysis assumes mu is zero.	11-15	I5
		=1, analysis assumes mu is not zero.		
	IEQ	=0, analysis assumes variances equal.	16-20	I5
		=1, analysis assumes variances not equal.		
	P	Spherical proportion requested.	21-30	F10.8
		($0 < P < 1$)		
	CCOEF	Confidence coefficient for confidence limits on *RP*.	31-35	F5.4
4		Data according to format FORMD.		
		Use records 5, 6, and 7 or 8 only if L1 = 0.		
5	IDIAG	=0, V is diagonal.	1-5	I5
		=1, V is not diagonal.		
	P	Spherical proportion requested.	6-15	F10.8
		($0 < P < 1$)		
6	MU	Mean vector elements MU(I), I = 1, 2, 3.	1-60	3F20.10
		Use record 7 if IDIAG = 0.		
7	V	Diagonal elements of V, V(I,I), I = 1, 2, 3.	1-60	3F20.10
		Use record 8 if IDIAG = 1.		
8	V	Elements of V, one row per line.		
		V(1,J), J = 1, 2, 3	1-60	3F20.10
		V(2,J), J = 1, 2, 3	1-60	3F20.10
		V(2,J), J = 1, 2, 3		

COMMENTS

A problem in SEPCL (estimation mode) is that of approximating *RP* with an analytical form which can subsequently be used to obtain confidence limits on *RP*. The problem can be stated as finding *RP* such that

$$\text{Prob}\left(\sum_i x_i^2 < (RP)^2\right) = P$$

under the trivariate normal assumptions on the x_i. To solve this problem analytically, we need the distribution of $\Psi^2 = \sum_i x_i^2$. It turns out that Ψ^2 is a weighted sum of non-central chi-square variables with a probability distribution which is too complicated to be useful. However, this distribution converges to a chi-square distribution as the σ_i^2 approach equality and the μ_i approach zero. Hence, the distribution of $\nu\Psi^2/E(\Psi^2)$ is approximated with a chi-square distribution with ν degrees of freedom. The degrees of freedom are obtained by the "method of matching moments," i.e., ν is the solution of the equation which equates the variance of $\nu\Psi^2/E(\Psi^2)$ with the variance of a chi-square with ν degrees of freedom. The solution to this equation and the results of the approximation are shown below:

$$(RP)^2 = K\left(\sum_i \sigma_i^2 + \sum_i \mu_i^2\right)/3$$

where

$$K = 3\chi_{\nu,P}^2/\nu\,.$$

In this expression, $\chi_{\nu,P}^2$ is the 100Pth percentile of a chi-square distribution with ν degrees of freedom where

$$\nu = 2m^2/c$$

and where m and c are the functions of the μ_i and σ_i^2 below:

$$m = 1 + \sum_i \mu_i^2 / \left(\sum_i \sigma_i^2\right)$$

$$c = 2\left(\sum_i \sigma_i^4 + 2\sum_i \sigma_i^2\mu_i^2\right) / \left(\sum_i \sigma_i^2\right)^2\,.$$

The user is referred to Reference 4 for the complete derivation.

In the estimation mode, the value *RP* is actually an estimate of the true *RP*. Using the circumflex (or hat) convention to denote estimates of parameters, one would write

$$(\hat{RP})^2 = K\left(\sum_i \hat{\sigma}_i^2 + \sum_i \hat{\mu}_i^2\right)/3\,.$$

To place confidence limits on *RP*, the distribution of

$$\nu_2(\hat{RP})^2/(RP)^2$$

is approximated with a chi-square distribution with ν_2 degrees of freedom using the "method of matching moments" described above. It is found that ν_2 has value

$$\nu_2 = 2n^2m^2/\left(nc + 2\sum_i \sigma_i^4/\left(\sum_i \sigma_i^2\right)^2\right).$$

The approximate 100 % confidence limits for *RP* are then taken as

$$\left(\frac{\hat{RP}}{\left[\chi^2_{\nu_2,\frac{1+\gamma}{2}/\nu_2}\right]^{1/2}}, \frac{\hat{RP}}{\left[\chi^2_{\nu_2,\frac{1-\gamma}{2}/\nu_2}\right]^{1/2}}\right).$$

The user is referred to Reference 6 for the complete derivation of *RP* confidence limits and an evaluation of their accuracy.

A brief discussion regarding the rotation of non-diagonal covariance matrices will aid the user in understanding his output. SEPCL is applicable to any problem where the errors are trivariate normal whether **V** is or is not diagonal. However, the routines for computing *RP* (exact or approximate) require that **V** be diagonal, i.e., that the x_i be independent random variables. Hence, if **V** is not diagonal on input (parameter input or an estimate from input data), a rotation must be performed which removes the off-diagonal elements from **V** and adjusts the diagonal elements accordingly. This process is based on the following principles: If the matrix **V** is symmetric there exists an orthogonal matrix **W** such that **W′VW** = **D** where **D** is diagonal (Reference 2). (An orthogonal matrix **W** is square and defined such that **WW′** = **I**.) Also, if the random vector **x** has a multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix **V**, i.e., if

$$\mathbf{x} \sim N(\boldsymbol{\mu}, \mathbf{V}),$$

and if **y** = **W′x**, then it follows from normal theory (Reference 1) that

$$\mathbf{y} \sim N(\mathbf{W}'\boldsymbol{\mu}, \mathbf{W}'\mathbf{V}\mathbf{W}).$$

Hence, if **x** is multivariate normal, it follows that there exists an orthogonal **W** such that **y** = **W′x** is multivariate normal with mean **W′**$\boldsymbol{\mu}$ and diagonal covariance matrix **D** = **W′VW**. Furthermore, since **W** is orthogonal, $\sum_i y_i^2 = \sum_i x_i^2$ so that

$$\text{Prob}\left(\sum_i x_i^2 < (RP)^2\right) = \text{Prob}\left(\sum_i y_i^2 < (RP)^2\right).$$

The result is that given **W**, one can transform from **x** to **y** (thus obtaining a new mean and a new diagonal covariance matrix), and the integral over any circular region is not affected by the transformation. The orthogonal matrix **W** is constructed in SEPCL by finding the eigenvalues of **V** and the eigenvector associated with each eigenvalue, and then taking **W** as the matrix of eigenvectors.

MISCELLANEOUS STATISTICAL ANALYSIS

LD50EST

PURPOSE

Program LD50EST (Lethal Dose 50 Estimation) computes estimates of the mean and standard deviation of an assumed normal distribution based on quantal response data. Quantal response data refers to a situation where a stimulus is applied to a test unit and the response is either a success or failure, a go or a no-go, etc. Examples of quantal response experimentation are found in experimental areas such as explosive sensitivity (drop tests, fragment impact tests, etc) and chemical sensitivity (insecticide tests, drug tests, etc.). The latter area originated the term Lethal Dose 50 (LD50) as a synonym for the median (and mean) of the assumed normal distribution for the population under consideration.

The examples mentioned fall into a special category of statistics called sensitivity analysis. Sensitivity analysis is based upon certain assumptions with respect to the testing environment. Each test unit is assumed to be associated with a critical stimulus level. When a unit is subjected to a stimulus less then its critical level it does not respond (failure). Conversely, when a unit is subjected to a stimulus greater than its critical level it responds positively (success). The distribution of critical stimulus levels of items from a particular population is assumed to be normal with mean, μ (LD50), and standard deviation, σ. Response data obtained over several stimulus levels is analyzed by the method of maximum likelihood to produce estimates, $\hat{\mu}$ (LD$\hat{50}$) and $\hat{\sigma}$, of the population parameters. A modified Newton-Raphson procedure is used to iteratively solve the maximum likelihood equations for the estimated values. Two requirements must be satisfied by the data in order for the solution procedure to obtain estimates of the parameters. The first requirement is that the average of the stimulus values which yielded successes must exceed the average of the stimulus values which yielded failures. The second requirement is that a zone of mixed results (ZMR) must exist. A ZMR exists when the maximum stimulus level associated with a failure exceeds the minimum stimulus level associated with a success. The variance-covariance matrix for $\hat{\mu}$ and $\hat{\sigma}$ is also computed by the program. The program computes point estimates and approximate confidence intervals for stimulus levels associated with user-specified probabilities of a success (positive response).

References 2 and 3 present the early development of statistical methods for analysis of sensitivity data. Reference 1 presents the formulation used in the original computer program and some statistical results from several sample data sets. The current STATLIB version of the program has been modified from the version presented in Reference 1. These modifications include the addition of computations for point and interval estimation of stimulus levels associated with specified success probabilities and the deletion of plotting options for simultaneous confidence ellipses for μ and σ.

FEATURES

The output features of LD50EST include the following:

* Printout of the stimulus levels and the number of "successes" and "failures" at each level.
* Estimates of the mean, $\hat{\mu}(\hat{LD50})$, and the standard deviation, $\hat{\sigma}$, of the assumed normal distribution of critical stimulus levels.
* Variance-covariance matrix for $\hat{\mu}(\hat{LD50})$ and $\hat{\sigma}$.
* Point estimation and approximate interval estimation of stimulus levels for user-specified success probability values and confidence coefficient.

REFERENCES

1. DiDonato, A. R. and Jarnigan, M. P., Jr. (1972), ***Use of the Maximum Likelihood Method under Quantal Responses for Estimating the Parameters of a Normal Distribution and Its Application to an Armor Penetration Problem***, NWL Technical Report TR-2846, NSWC, Dahlgren, VIRGINIA 22448.
2. Dixon, W. J. and Mood, A. M. (1948), "A Method for Obtaining and Analyzing Sensitivity Data", Journal of the American Statistical Association, Vol. 43, pp. 109 - 126.
3. Golub, A. and Grubbs, F. (1956), "Analysis of Sensitivity Experiments when the Levels of the Stimulus Cannot be Controlled", Journal of the American Statistical Association, Vol. 51, pp. 257 - 265.

INPUT GUIDE

The specifications of the user-created input file are given below. The file must contain record types 1, 2, 3, 4, 5A and 5B for analysis of a single data set. Multiple data sets can be processed with a single input file by consecutively including corresponding sets of record types 1 through 5 (A and B).

Record Type	Variable	Description	Columns	Format
1	IDENT	Problem description.	1-72	9A8
2	NL	Number of unique stimulus levels where positive responses ("successes") occurred.	1-4	I4
	ML	Number of unique stimulus levels where negative responses ("failures") occurred.	5-8	I4

	INPUT	=0, default starting values for the iterative solution process will be computed for estimates of alpha and beta. See COMMENTS. =1, starting values for the iterative solution process will be input for estimates of alpha and beta. See COMMENTS.	9-10	I2
	ITE	=0, no printout of intermediate estimates of alpha and beta at each iteration of the solution. =1, intermediate estimates of alpha and beta at each iteration of the solution are printed.	11-12	I2
	ALPHA0	Starting value for alpha.	13-32	E20.14
	BETA0	Starting value for beta.	33-52	E20.14
	CCOF	Confidence coefficient for interval estimation of stimulus levels. $(0 < CCOF < 1)$	53-60	F8.4
3	NPL	Number of success probability values for which point and interval estimation is requested. $(1 \leq NPL \leq 15)$	1-5	I5
	PL(I)	Success probability values for point and interval estimation of stimulus levels. NPL values must be entered.	6-80	15F5.3
4	FORM	Format (in parentheses) for reading response data. The response data consists of two attributes at each stimulus level, the number of responses ("successes" or "failures") and associated stimulus level. The number of responses must be read as an integer variable. Example format: (10(I2,F6.1))	1-80	10A8
5A	FNA(i), A(i) i=1,...,NL	Enter response data for "successes". Data consists of NL pairs of numbers. The first value is the number of "successes" and the second value is the associated stimulus level. The number of "successes" must be in integer format. Repeat this record type as required to enter all "success" data.	See FORM	FORM

5B	FNB(i), B(i) i=1,...,ML	Enter response data for "failures". Data consists of ML pairs of numbers. The first value is the number of "failures" and the second value is the associated stimulus level. The number of "failures" must be in integer format. Repeat this record type as required to enter all "failure" data.	See FORM	FORM

COMMENTS

The parameters μ(LD50) and σ of the assumed normal distribution of critical stimulus levels are not estimated directly by the procedure given in Reference 1. A transformation is made to the variables,

$$\alpha = \mu/\sigma \text{ and } \beta = 1/\sigma,$$

as a simplification to the iterative Newton-Raphson solution procedure. Therefore, $\hat{\alpha}$ and $\hat{\beta}$ are obtained directly by the program and retransformed to the estimates $\hat{\mu}$ and $\hat{\sigma}$. Starting values for $\hat{\alpha}$ and $\hat{\beta}$ are required by the solution procedure and must be input by the user or computed by the program under the default algorithm. Experience has shown that the default algorithm generates satisfactory values in almost every case.

For data sets consisting of stimulus levels near zero, it is possible to obtain estimates of μ(LD50) which are negative if the ZMR is relatively large. In these cases, the negative value of the estimate is printed parenthetically and a value of zero is printed as a "working" estimate of μ(LD50). In subsequent calculations for point and interval estimation of stimulus levels for selected success probabilities, the value of zero is used for $\hat{\mu}$(LD50) Any negative estimate for a point or interval bound is printed as a zero in the output. The foregoing is based on the assumption that only positive stimulus values are possible. Although the solution algorithm will process negative values, a situation with both positive and negative stimulus levels does not readily come to mind. Consequently, the user is advised to restrict the input stimulus levels to positive values and to treat zero-value estimates cautiously.

FFAC2K

PURPOSE

Program FFAC2K (Fractional Factorial Experiments with 2 levels of each of k factors) computes estimates of the factorial effects and the analysis of variance (anova) table for factorial experiments of the 2^k configuration. Factorial experiments involve a particular arrangement of factor level combinations in which each level of every factor is crossed with each level of every other factor. Factorial experiments with k factors at each of two levels are quite common in experimental design and are placed in the special category of 2^k experiments. The classical "Yates Procedure" (References 1 and 2) is used to perform the computations and an optional computational check on these results is available to the user. Up to 10 factors may be considered for each full factorial experiment.

The program also has the capability to provide for analysis of fractional replication of factorial experiments of the 2^k type. Fractional replication of a factorial experiment requires less experimentation than the corresponding full factorial. However, this results in confounding or aliasing of factorial effects in the analysis of variance table. Confounding (aliasing) means that two or more effects are estimated by the same arithmetic function of the response data. In order to design or analyze fractional experiments one must be aware of the alias structure of the factorial effects. The program FFAC2K can generate the alias structure based on the user supplied fundamental identity (sometimes called a defining contrast) associated with a particular fractional design.

References 1, 2 and 3 include discussions of factorial and fractional factorial experiments and their analyses. Many designs of fractional factorials of the 2^k configuration are given in Reference 4.

FEATURES

The output features of FFAC2K include the following:

* Printout of the input observations of the response variable and the associated factor level combination.
* Optional computational check on the Yates Procedure.
* Printout of the analysis of variance table including for each factorial effect, the estimated effect, the sum of squares, the F statistic and the probability level at which the effect is significant.
* The sum of squares, degrees of freedom and mean square for the experimental error.
* The defining contrast and alias structure as optional printout.

REFERENCES

1. Box, G. E. P., Hunter, W. G. and Hunter, J. S. (1978), ***Statistics for Experimenters***, John Wiley & Sons, pp. 306 - 432.
2. Cochran, W. G. and Cox, G. M. (1957), ***Experimental Design***, John Wiley & Sons, Inc., pp. 148 - 161 and pp. 244 - 270.
3. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan Publishing Co., pp. 503 - 528.
4. National Bureau of Standards Applied Mathematics Series*48 (1957), ***Fractional Factorial Experiment Designs for Factors at Two Levels***, U. S. Government Printing Office.

INPUT GUIDE

The specifications of the user-created input file are given below. Record types 1, 2, 3 and either 4 or 5 are mandatory. Record type 4 contains the observations of the response variable and must be included if the program is being used for analysis of either a full or fractional factorial experiment. Record type 5 contains the defining contrast and must be included if the alias structure of a fractional experiment is desired. If both the analysis and alias structure are desired than both record types 4 and 5 must be included.

Record Type	Variable	Description	Columns	Format
1	ID	Problem description.	1-72	9A8
2	NFAC	Number of experimental factors. In a full 2^k factorial this value is k. For a fractional factorial, such as a (1/2)**m replicate of a 2^k factorial, this value is k-m. See the COMMENTS section. (NFAC ≤ 10)	1-2	I2
	NREP	Number of replications of the full or fractional factorial experiment. (NREP ≤ 100) If only the alias structure is desired, NREP must be 0 or blank and ALS (columns 7-8) must be 1. In this case the defining contrast (record type 5) must be included and the response data (record type 4) must be omitted.	3-6	I4

	ALS	=0, Alias structure will not be generated with the analysis. The defining contrast (record type 5) must be omitted.	7-8	I2
		=1, Alias structure will be generated with the analysis. The defining contrast (record type 5) must be included. If only the alias structure is desired (NREP = 0), a one must be entered here.		
	NYP	=0, Computational check on the Yates Procedure will not be done.	9-10	I2
		=1, Computational check on the Yates Procedure will be done.		
	NFN	The number, m, in the expression, (1/2)**m, which represents the amount of fractionation. For example, enter 0 for full replication: (1/2)**0=1; 1 for 1/2 replicate: (1/2)**1=1/2; 2 for 1/4 replicate: (1/2)**2=1/4; etc.	11-12	I2
	FMT	Format (in parentheses) for reading response data.	17-80	8A8
3	FAC(1)	Letter assigned to the first factor.	1	A1
	FAC(2)	Letter assigned to the second factor.	2	A1
...	.....	.....	....	...
	FAC(10)	Letter assigned to the tenth factor.	10	A1

The NFAC letters must be entered in alphabetical order. NFAC ≤ 10. See the COMMENTS section with respect to assignment of letters to factors in the case of fractional replication.

If NREP > 0 (on record type 2), record type 4 must be included.

4	T(j), j=1,..., NREP	Enter the data observations according to FMT. Data must be in the "standard order" in the context of factorial experiments. See COMMENTS section. In the case of multiple replications, all observations from the same factor level combination must be entered consecutively. Then the observations for the factor level combination next in line with respect to the "standard order" are entered, etc.	FMT

If NREP = 0 and ALS = 1 (on record type 2), record type 5 must be included.

5	NDC	Number of effects in the defining contrast. See COMMENTS section for discussion of effect and defining contrast.	1-3	I3
	DEFCON(1)	First effect of the defining contrast (left justified).	5-20	2A8
	DEFCON(2)	Second effect of the defining contrast (left justified).	22-37	2A8
	DEFCON(3)	Third effect of the defining contrast (left justified).	39-54	2A8
	DEFCON(4)	Fourth effect of the defining contrast (left justified).	56-71	2A8

If NDC > 4, additional effects are indicated on subsequent records according to the format (2A8, 3(1X,2A8)), i.e., fifth, sixth, seventh and eighth effects beginning in columns 1, 18, 35 and 52, respectively, etc. Within each effect the factor letters must be arranged alphabetically, e.g., AB rather than BA. All effects must be left justified.

COMMENTS

In 2^k factorial experiments, k factors, each at two levels, are crossed with each other to produce 2^k factor level combinations. Upper case letters are used to name the factors. Factor level combinations are designated by using corresponding lower case letters. The presence of a lower case letter in the designation of a factor level combination indicates the corresponding factor is at the "high" level. The lower case letter is omitted from the designation if the factor is at the "low" level.

Consider an experiment with four factors represented by A, B, C and D. There are 2^4 = 16 factor level combinations in this experiment. In the "standard order" of these 16

combinations the first combination has all factors at the "low" level and is conventionally designated by (1). The second combination is obtained by introducing factor A at the "high" level. Multiplying the first factor level combination by factor A at the "high" level (1 × a) generates the second factor level combination, a, in the "standard order". (For combination "a" all factors are at the "low" level except factor A.) The next two factor level combinations are obtained by introducing factor B at the "high" level. Multiplying combinations one and two by factor B at the "high" level (1 × b and a × b) generates combinations b and ab. Proceeding similarly with factor C, multiplying combinations (1), a, b and ab each by c generates combinations c, ac, bc and abc. After completing the generation scheme with factor D, the complete list of factor level combinations in "standard order" becomes (1), a, b, ab, c, ac, bc, abc, d, ad, bd, abd, cd, acd, bcd and abcd. The response data must be entered into the program in this order.

For fractional factorial experiments the number of treatment combinations is reduced by the amount of fractionation. For example, in a 1/2 replicate of a 2^5 experiment, the number of factor level combinations performed will be $1/2 \times 2^5 = 2^{5-1} = 2^4 = 16$, rather than $2^5 = 32$, the number required by a non-fractionated experiment. Effectively, this reduces the experiment to the equivalent of a pseudo 2^4 full factorial experiment. When using the program to analyze fractional data, one must consider this reduction of factor level combinations in the preparation of the input file. Obviously, the input parameters NFAC and FAC(i) are affected but the "standard order" of factor level combinations is also affected, as is discussed below.

Using a half-replicate of a 2^4 experiment with the four-factor interaction effect, ABCD, as the defining contrast for an example, NFAC = 3 and FAC(1), FAC(2) and FAC(3) equal A, B and C, respectively. The 16 treatment combinations of a full 2^4 factorial were listed in "standard order" above. Only 8 of these treatment combinations would be performed in a half-replicate of a 2^4 experiment. By use of the defining contrast the 16 combinations can be divided into two half-replicates of 8 combinations each. For the interested user the algorithm for doing this is given in Reference 2 on pages 247 - 253. The analyst does not have to apply the algorithm, however, because the appropriate factor combinations to be performed are given in the design references such as Reference 4. The two half-replicates for the example are given below:

Half-rep 1: a, b, c, abc, d, abd, acd, bcd

Half-rep 2: (1), ab, ac, bc, ad, bd, cd, abcd

Either set is a candidate for a fractional factorial. However, in both sets a reordering must take place to conform to the "standard order" required by the program. Because the analysis is being done as a full replicate of a pseudo 2^3 experiment with factors A, B and C, the factor level combinations must be reordered ignoring factor D. Therefore, the "standard order" for each of the half-replicates becomes:

Half-rep 1: (d), a, b, ab(d), c, ac(d), bc(d), abc

Half-rep 2: (1), a(d), b(d), ab, c(d), ac, bc, abc(d)

The factor level designator (d) has been placed in parentheses to indicate that it should not be considered in the construction of the "standard order".

Some additional remarks are necessary with respect to the defining contrast. In a full 2^k factorial there are 2^k - 1 experimental effects that can be estimated. These effects can be designated by factor letters and combinations of factor letters. In the example of a full 2^4 factorial the 15 effects are the main effects, A, B, C and D; the two-factor interactions, AB, AC, AD, BC, BD and CD; the three-factor interactions, ABC, ABD, ACD and BCD; and the four-factor interaction, ABCD. To define a half-replicate one of these effects must be chosen as the defining contrast and in the example, the ABCD effect was selected. The alias structure can then be generated by obtaining the "generalized interaction" of each experimental effect with the defining contrast. The generalized interaction of two effects is the algebraic product of their letter designations where the exponents are reduced to the modulo 2 base. The alias of the effect A in the example is its generalized interaction with ABCD which is A^2BCD or simply BCD. The alias of BCD, obviously, is A which is equivalent to the generalized interaction $AB^2C^2D^2$.

For fractionation beyond half-replication, the generalized interaction is useful in identifying all effects in the defining contrast. For example, consider a quarter-replicate of a 2^6 factorial with factors A, B, C, D, E and F. In order to subdivide the $2^6 = 64$ treatment combinations into four quarter-replicates each containing 16 combinations, two effects must be chosen for the defining contrast. However, a third effect, defined by the generalized interaction of the chosen two effects, automatically becomes a part of the defining contrast. If ABCE and ABDF are selected as the defining contrast then A^2B^2CDEF = CDEF also becomes part of the defining contrast.

Additionally, care must be taken when assigning letters to factors on Record Type 3 for fractional replication. The assignment of letters must be such that it does not produce a factorial effect name which is identical with any of the effects comprising the defining contrast. This would generate an incomplete alias structure and, therefore, is flagged by the program. In this instance the user is advised to consider a different letter assignment or a different defining contrast.

An example may help the user understand this restriction on naming the factors. Consider a 1/4 replicate of a 2^5 factorial to be analyzed as a pseudo 2^3 factorial. Assume a letter assignment of (A, B, C) for the three factors of the pseudo 2^3 factorial and (D, E) for the two remaining factors introduced by the defining contrast of

$$I = ABDE = BCDE = AC.$$

Without the flag the program would generate the following incomplete alias structure in terms of the seven factorial effects from the pseudo 2^3 factorial:

A = BDE = ABCDE = C AB = DE = ACDE = BC
B = ADE = CDE = ABC AC = BCDE = ABDE = I
C = ABCDE = BDE = A BC = ACDE = DE = AB
ABC = CDE = ADE = B

Several alias relationships are missing from this alias structure while others are duplicated. In this situation the user is instructed to redo the input file. The user may choose to rename the factors or specify a different defining contrast. A suggested choice for this example is to retain A and B while replacing C with D for the three factors of the pseudo 2^3 factorial. The factors C and E are then introduced by the given defining contrast. This naming convention in conjunction with the same defining contrast yields the following alias structure:

A = BDE = ABCDE = C AB = DE = ACDE = BC
B = ADE = CDE = ABC AD = BE = ABCE = CD
D = ABE = BCE = ACD BD = AE = CE = ABCD
ABD = E = ACE = BCD

The original alias relationships have been retained while providing for the previously missing ones to be generated.

SUBROUTINES

RANDOM NUMBER GENERATION

The property of randomness is a key element in many areas of scientific research and application. Random numbers generated on a digital computer are used in several ways in addition to sampling from specific distributions or populations. Other uses include simulation studies in which the behavior of a system that contains random components is modeled and computer program checkout in which combinations of input parameters used to test the code are randomly selected.

When speaking of random numbers one is usually referring to a sequence of numbers which obeys some probability law; for example, the sequence $X_1, X_2,..., X_n$ might represent n independent numbers drawn from a continuous uniform distribution over the interval (a,b). These U(a,b) numbers are referred to as "uniform random numbers" or "uniform random variates." The need for random numbers makes the availability of a readily accessible set of algorithms for generating random numbers from a wide variety of probability distributions on a digital computer quite desirable. STATLIB provides such a facility through inclusion of a set of twenty-four subroutines for random number generation from both discrete and continuous distributions. While this set is not exhaustive, it does include virtually all of the common distributions required for most user purposes. The basis for the generation scheme used in each of these subroutines is an algorithm for generating a sequence of independent uniform random variates on the interval (0,1). Probability theory establishes the fact that variates can be generated from a large number of distributions provided that such a sequence can be generated. The random number subroutines in STATLIB employ subroutine URNG, currently available on the CDC system in the MATHLIB library, to produce the required U(0,1) variates.

Documentation on each of the random number generation subroutines in STATLIB follows in the ensuing pages. Each subroutine is designed to enable the user to generate a set of random variates from the desired distribution. In the description of each subroutine the functional form of the probability distribution in question is displayed whenever deemed appropriate. Oftentimes there is more than one accepted functional form in the literature. In such cases only one form has been selected and displayed. In addition, the mean and variance of each distribution is given in most cases to allow the user to check the sample mean and variance from the generated set of random variates against their theoretical values if so desired.

In the call line for each subroutine FORTRAN naming conventions have been followed for all variables. Hence, variables whose first letter is in the range I to N are integers while all other variables are real. In the input guide for each subroutine the call line is listed twice, once on a single line and once in currently accepted structured programming form. For example, the call line for subroutine RANUWO is CALL RANUWO (N,IA,IB,K,ISEED,IC,IX,IERROR), while in structured programming form it appears as

```
    CALL RANUWO
G               ( N      , IA      , IB      , K
B               , ISEED  , IC
Y               , IX     , IERROR )
```

This structure is used in order to facilitate the identification of those arguments for which the user must provide input values. There are three categories of arguments: given (G), both (B), and yielded (Y). The given (G) arguments require input values provided by the user. These argument values will not change during execution. The yielded (Y) arguments identify output values returned by the subroutine. Hence, they represent created values and as such require no user action. The both (B) category refers to arguments whose values are modified during execution of the subroutine. Hence, these arguments may or may not require the user to input values. The input guide provided with each subroutine description clarifies which of these arguments require input and which do not. For example, in subroutine RANUWO a value for the argument ISEED must be input by the user, while no input is required for the argument IC. IC is an array of dimension K whose values are initialized to zero within RANUWO and then updated repeatedly before execution is completed. IC is included in the argument list so that its size does not have to be restricted in the subroutine dimension statement.

Two arguments which are common to all of the random number generation subroutines require some discussion. The last argument in each subroutine call line is an input error flag. When a subroutine is called, if no input errors are detected then this argument is set to zero. If an input error is detected, the value of this argument indicates which specific input error has been made. The argument ISEED is, on input, an integer "seed" used to initialize a sequence of U(0,1) variates generated by subroutine URNG. Recall that each random number generation subroutine in STATLIB utilizes these U(0,1) variates in its generation scheme. On output, ISEED is a new seed available for generating additional U(0,1) variates from subroutine URNG. The input value for ISEED must be such that $1 \leq ISEED < 2^{31} - 1$. A given value of ISEED always initiates the same set of U(0,1) variates. Note that it is feasible to specify only one input value of ISEED in a main program in order to generate successive blocks of random variates from different STATLIB subroutines. For example, suppose a main program requires 25 gamma random variates followed by 50 normal random variates. One possible set of instructions would be

```
ISEED = 5437
CALL RANGAM (25,A,B,ISEED,X,IERRORX)
CALL RANNOR (50,FMU,SIG,ISEED,Y,IERRORY)
```

The value of ISEED is updated each time subroutine URNG is called by subroutines RANGAM and RANNOR. The last of these updated values from subroutine RANGAM will be the first ISEED value used by subroutine RANNOR.

One final comment regarding random number generation is in order. The sequence of U(0,1) variates used in generating variates from other distributions are themselves produced using the arithmetic operations of a digital computer in a recursive scheme. (Subroutine URNG is based on a multiplicative congruential generation scheme.) As such, once the seed value is specified, the U(0,1) sequence is completely determined. Such sequences are not really random, but they appear to be, and are in fact referred to as pseudo-random. They are called random with the above understanding. For typical applications these deterministic sequences may be considered random if they satisfy certain statistical properties of randomness. Subroutine URNG has been thoroughly tested and clearly exhibits all of the critical randomness properties. It turns out that the random number generation subroutines included in STATLIB, all of which utilize U(0,1) numbers from subroutine URNG, also satisfy these randomness properties.

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 167 - 241.
2. Hull, T. E. and Dobell, A. R. (1962), "Random Number Generators", SIAM Review, 4, pp. 230 - 254.
3. Jansson, B. (1966), ***Random Number Generators***, Almqvist and Wiksell.
4. Knuth, D. E. (1981), ***The Art of Computer Programming***, Second Edition, Addison-Wesley, Volume 2 / Seminumerical Algorithms, pp. 1 - 177.
5. MacLaren, M. D. and Marsaglia, G. (1965), "Uniform Random Generators", Journal of the Association of Computing Machinery, 12, pp. 83 - 89.

DISCRETE

RANDOM NUMBER

GENERATORS

RANARB

PURPOSE

Subroutine RANARB generates n random variates from an arbitrarily specified discrete distribution with replacement. The user specifies the values of the random variable and their associated probabilities of occurrence. If a discrete distribution assumes the values $x_1, x_2, \ldots, x_n$ with corresponding probabilities $p_1, p_2, \ldots, p_n$ where $\sum_{i=1}^{n} p_i = 1$, then the form of the distribution is

$$p(x_i) = p_i \quad , \quad i = 1,2,\ldots,n$$

The values of the random variable may be integer and/or real, but they are treated as reals in RANARB.

FEATURES

The input arrays of variate values and their associated probabilities are simultaneously reordered in RANARB to make the generation scheme computationally efficient.

REFERENCE

1. Knuth, D. E. (1981), ***The Art of Computer Programming***, Second Edition, Addison-Wesley, Volume 2 / Seminumerical Algorithms, p. 115 and Volume 1 / Fundamental Algorithms, pp. 399 - 404.

INPUT GUIDE

The call line for subroutine RANARB is

```
CALL RANARB (N,K,X1,P,ISEED,X,XL,IERROR)
```

or, in structured programming form,

```
     CALL RANARB
G                 ( N       , K
B                 , X1      , P       , ISEED
Y                 , X       , XL      , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated
($N > 0$)

K : Number of values that the discrete random variable can assume
($K > 0$)

Both

X1(K) : Input array of dimension K containing the values of the discrete random variable

P(K) : Input array of dimension K containing the probabilities with which the discrete random variable can assume each of its K possible values
($0.0 \leq P(I) \leq 1.0, I = 1,K$)

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N) : Output array of dimension N containing the generated variates

XL(K) : Auxiliary array of dimension K containing cumulative sums of the P(I)'s used in the generation scheme

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Number of variate values out of range
3 : At least one probability out of range
4 : Probabilities fail to sum to unity
5 : Seed out of range)

COMMENTS

The user must take care to ensure that the input probabilities sum to one to yield a valid probability distribution.

RANBER

PURPOSE

Subroutine RANBER generates n Bernoulli random variates (i.e., 0's and 1's) from a Bernoulli distribution with parameter p (probability of success on a single trial). The form of the Bernoulli distribution used is

$$p(x) = p^x(1-p)^{1-x}, x = 0, 1$$

with

$$\text{mean} = p$$

and

$$\text{variance} = p(1\text{-}p)$$

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 168 - 169.

INPUT GUIDE

The call line for subroutine RANBER is

```
CALL RANBER (N,P,ISEED,IX,IERROR)
```

or, in structured programming form,

```
    CALL RANBER
G               ( N         , P
B               , ISEED
Y               , IX        , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated (N > 0)

P : Probability of success on a single trial
($0.0 \leq P \leq 1.0$)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

IX(N) : Output array of dimension N containing the generated Bernoulli variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Probability of success out of range
3 : Seed out of range)

RANBIN

PURPOSE

Subroutine RANBIN generates *n* binomial random variates from a binomial distribution with parameters *nt* (number of trials) and *p* (probability of success on a single trial). The form of the binomial distribution used is

$$p(x) = \binom{nt}{x} p^x (1-p)^{nt-x}, x = 0, 1, \ldots, nt$$

The mean and variance of the binomial are

$$\text{mean} = nt \cdot p$$

$$\text{variance} = nt \cdot p(1-p)$$

FEATURES

The generation scheme used in RANBIN is in three parts. If the number of trials (*nt*) is less than or equal to 100, the generation scheme is based on summing Bernoulli variates. If *nt* is greater than 100, the generation scheme is based on either the normal approximation to the binomial or the Poisson approximation to the binomial. If $nt \cdot p$ is greater than 5 and *p* is less than or equal to 0.5 or if $nt(1 - p)$ is greater than 5 and *p* is greater than 0.5, the normal approximation is used. Otherwise the Poisson approximation is employed.

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 220, pp. 211 - 213, and p. 224.
2. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, p. 126.

INPUT GUIDE

The call line for subroutine RANBIN is

```
CALL RANBIN (N,NT,P,ISEED,IX,IERROR)
```

or, in structured programming form,

```
      CALL RANBIN
G                 ( N       , NT      , P
B                 , ISEED
Y                 , IX      , IERROR )
```

The parameter list is given below:

Given arguments

- N : Number of variates to be generated ($N > 0$)
- NT : Number of trials ($NT \geq 0$)
- P : Probability of success on a single trial ($0.0 \leq P \leq 1.0$)

Both

- ISEED : Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

- IX(N) : Output array of dimension N containing the generated binomial variates
- IERROR : Input error flag
  - (0 : No input errors
    1 : Number of variates out of range
    2 : Number of trials out of range
    3 : Probability of success out of range
    4 : Seed out of range)

RANGEO

PURPOSE

Subroutine RANGEO generates n geometric random variates from a geometric distribution with parameter p (probability of success on a single trial). The form of the geometric distribution used is

$$p(x) = p(1-p)^{x-1} \quad , \quad x = 1,2,3,\ldots$$

with

$$\text{mean} = 1/p$$

and

$$\text{variance} = (1-p)/p^2$$

REFERENCE

1. Knuth, D. E. (1981), ***The Art of Computer Programming***, Volume 2 / Seminumerical Algorithms, Second Edition, Addison-Wesley, p. 131.

INPUT GUIDE

The call line for subroutine RANGEO is

```
CALL RANGEO (N,P,ISEED,IX,IERROR)
```

or, in structured programming form,

```
    CALL RANGEO
G               ( N         , P
B               , ISEED
Y               , IX        , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated (N > 0)

P : Probability of success on a single trial
($0.0 \leq P \leq 1.0$)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

IX(N) : Output array of dimension N containing the generated geometric variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Probability of success out of range
3 : Seed out of range)

RANHYP

PURPOSE

Subroutine RANHYP generates n hypergeometric random variates from a hypergeometric distribution with parameters NP (finite population size), M (number of successes in the finite population), and ns (number of items sampled from the finite population). The hypergeometric distribution is appropriate when sampling without replacement from a finite population. The form of the hypergeometric distribution used is

$$p(x) = \frac{\binom{M}{x}\binom{NP-M}{ns-x}}{\binom{NP}{ns}} \quad , x = \max(0, ns - NP + M), \ldots, \min(M, ns)$$

The mean and variance of the hypergeometric are

$$\text{mean} = (ns \cdot M)/NP$$

$$\text{variance} = (NP - ns)/(NP - 1) \cdot ns \cdot (M/NP) \cdot (1 - (M/NP))$$

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 228.

INPUT GUIDE

The call line for subroutine RANHYP is

```
CALL RANHYP (N,NP,M,NS,ISEED,IX,IERROR)
```

or, in structured programming form,

```
    CALL RANHYP
G               ( N       , NP      , M       , NS
B               , ISEED
Y               , IX      , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated (N > 0)
NP	:	Size of the finite population from which a sample of size NS is to be taken (NP > 0)
M	:	Number of success items in the finite population (0 < M ≤ NP)
NS	:	Numper of items to be sampled from the finite population (0 < NS ≤ NP)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

IX(N)	:	Output array of dimension N containing the generated hypergeometric variates
IERROR	:	Input error flag (0 : No input errors 1 : Number of variates out of range 2 : Finite population size out of range 3 : Number of success items out of range 4 : Sample size out of range 5 : Seed out of range)

RANNBI

PURPOSE

Subroutine RANNBI generates n negative binomial random variates from a negative binomial distribution with parameter p (probability of success on a single trial). The form of the negative binomial distribution used is

$$p(x) = \binom{x-1}{k-1} p^k (1-p)^{x-k} \quad , \quad \begin{matrix} x = k, k+1, k+2, \ldots \\ k = 1, 2, 3, \ldots \end{matrix}$$

The mean and variance of the negative binomial are

$$\text{mean} = k / p$$

$$\text{variance} = k(1-p)/p^2$$

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 226.
2. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, pp. 121 - 122.

INPUT GUIDE

The call line for subroutine RANNBI is

```
CALL RANNBI (N,P,K,ISEED,IX,IERROR)
```

or, in structured programming form,

```
    CALL RANNBI
G               ( N       , P       , K
B               , ISEED
Y               , IX      , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated
($N > 0$)

P : Probability of success on a single trial
($0.0 \leq P \leq 1.0$)

K : Number of successes required
($K > 0$)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

IX(N) : Output array of dimension N containing the generated negative binomial variates. Each IX value represents the number of the trial on which the Kth success occurs.

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Probability of success out of range
3 : Number of successes out of range
4 : Seed out of range)

RANPOI

PURPOSE

Subroutine RANPOI generates n Poisson random variates from a Poisson distribution with parameter *fmu* (*fmu* > 0). *fmu* represents the average rate per time (or space) interval. The form of the Poisson distribution used is

$$p(x) = \frac{e^{-fmu} fmu^{x}}{x!} \quad , \quad \begin{array}{l} x = 0, 1, 2, \ldots \\ fmu > 0 \end{array}$$

The mean and variance of the Poisson are

mean = *fmu*

variance = *fmu*

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 224.

INPUT GUIDE

The call line for subroutine RANPOI is

```
CALL RANPOI (N,FMU,ISEED,IX,IERROR)
```

or, in structured programming form,

```
    CALL RANPOI
G               ( N       , FMU
B               , ISEED
Y               , IX      , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated (N > 0)

FMU : Average rate per time (or space) interval
(FMU > 0)

Both

ISEED : Integer seed
($1 \leq$ ISEED $< 2^{31} - 1$)

Yielded arguments

IX(N) : Output array of dimension N containing the generated Poisson variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Poisson parameter FMU out of range
3 : Seed out of range)

RANUWO

PURPOSE

Subroutine RANUWO generates a randomly ordered subset of n of the integers in the interval *IA* to *IB* inclusive (*IA* ≤ *IB*). All integers in the resultant random ordering are distinct; that is, the integers are generated without replacement.

INPUT GUIDE

The call line for subroutine RANUWO is

CALL RANUWO (N,IA,IB,K,ISEED,IC,IX,IERROR)

or, in structured programming form,

```
      CALL RANUWO
G                 ( N       , IA        , IB        , K
B                 , ISEED   , IC
Y                 , IX      , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated ($0 < N \leq K$)
IA	:	Lower limit of generation interval ($IA \leq IB$)
IB	:	Upper limit of generation interval ($IB \geq IA$)
K	:	Length of generation interval ($K = IB - IA + 1$)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)
IC(K)	:	Auxiliary array of dimension K used for keeping track of previously generated integers

Yielded arguments

IX(N) : Output array of dimension N containing the generated integers

IERROR : Input error flag

(0 : No input errors
1 : Improper generation interval
2 : Incorrect generation interval width
3 : Number of variates out of range
4 : Seed out of range)

RANUWR

PURPOSE

Subroutine RANUWR generates *n* random integers in the range *IA* to *IB* inclusive (*IA* $\leq$ *IB*) with replacement. All integers in this range are assumed to be equally likely to occur; that is, generation is based on the discrete uniform distribution. The form of the discrete uniform distribution used is

$$p(x) = 1 / (IB - IA + 1) \quad , x = IA, IA+1, IA+2, \ldots , IB$$

The mean and variance of the discrete uniform are

$$\text{mean} = IA + (IB - IA)/2$$

$$\text{variance} = ((IB - IA + 1)^2 - 1)/12$$

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 219.

INPUT GUIDE

The call line for subroutine RANUWR is

```
CALL RANUWR (N,IA,IB,ISEED,IX,IERROR)
```

or, in structured programming form,

```
    CALL RANUWR
G                   ( N       , IA        , IB
B                   , ISEED
Y                   , IX      , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated (N > 0)

IA : Lower limit of generation interval
(IA ≤ IB)

IB : Upper limit of generation interval
(IB ≥ IA)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

IX(N) : Output array of dimension N containing the generated discrete uniform variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Improper generation interval
3 : Seed out of range)

CONTINUOUS

RANDOM NUMBER

GENERATORS

RANBET

PURPOSE

Subroutine RANBET generates n beta random variates from a beta distribution with parameters a and b (a and b both > 0). The form of the beta distribution used is

$$f(x) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1}(1-x)^{b-1} \quad , \quad \begin{array}{l} 0 < x < 1 \\ a > 0 \\ b > 0 \end{array}$$

The mean and variance of the beta distribution are

$$\text{mean} = a/(a+b)$$

$$\text{variance} = ab/((a+b)^2(a+b+1))$$

FEATURES

The generation scheme used in RANBET is written in three parts, keying upon whether the parameters a and b are integral or nonintegral as well as their magnitudes, to increase computational efficiency.

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 204 - 208 and pp. 209 - 211.

INPUT GUIDE

The call line for subroutine RANBET is

```
CALL RANBET (N,A,B,ISEED,X,IERROR)
```

or, in structured programming form,

```
      CALL RANBET
G                 ( N       , A       , B
B                 , ISEED
Y                 , X       , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated
($N > 0$)

A : Parameter of the beta distribution
($A > 0$)

B : Parameter of the beta distribution
($B > 0$)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N) : Output array of dimension N containing the generated beta variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Beta parameter A out of range
3 : Beta parameter B out of range
4 : Seed out of range)

RANCSQ

PURPOSE

Subroutine RANCSQ generates *n* chi-square random variates from a chi-square distribution with parameter *nu* (called the degrees of freedom of the distribution) where *nu* is a positive integer. The form of the chi-square distribution used is

$$f(x) = \frac{1}{2^{nu/2}\Gamma(nu/2)} x^{\frac{nu}{2}-1} e^{-\frac{x}{2}} \quad , \quad \begin{array}{l} 0 < x < \infty \\ nu = 1, 2, 3, \ldots \end{array}$$

The mean and variance of the chi-square distribution are

mean = *nu*

variance = $2 \cdot nu$

FEATURES

The generation scheme used in RANCSQ is based on the relationship between the gamma and chi-square distributions; namely, the chi-square with *nu* degrees of freedom is equivalent to a gamma distribution with parameters $a = nu/2$ and $b = 2$.

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 213.
2. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, pp. 155 - 156.

INPUT GUIDE

The call line for subroutine RANCSQ is

CALL RANCSQ (N,NU,ISEED,X,IERRORS)

or, in structured programming form,

```
      CALL RANCSQ
G                  ( N       , NU
B                  , ISEED
Y                  , X       , IERRORS )
```

The parameter list is given below:

Given arguments

- N : Number of variates to be generated
  ($N > 0$)
- NU : Degrees of freedom parameter of the chi-square distribution
  ($NU > 0$)

Both

- ISEED : Integer seed
  ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

- X(N) : Output array of dimension N containing the generated chi-square variates
- IERRORS : Input error flag
  (0 : No input errors
  1 : Number of variates out of range
  2 : Degrees of freedom parameter out of range
  3 : Seed out of range)

RANEXP

PURPOSE

Subroutine RANEXP generates n exponential random variates from an exponential distribution with parameter a ($a > 0$). a is interpreted as the average rate per unit of time or the average time to failure (average lifetime). The form of the exponential distribution used is

$$f(x) = \frac{1}{a} e^{-x/a} \quad , \quad \begin{matrix} x > 0 \\ a > 0 \end{matrix}$$

The mean and variance of the exponential are

mean = a

variance = a^2

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 203.

INPUT GUIDE

The call line for subroutine RANEXP is

```
CALL RANEXP (N,A,ISEED,X,IERROR)
```

or, in structured programming form,

```
    CALL RANEXP
G               ( N       , A
B               , ISEED
Y               , X       , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated (N > 0)

A : Mean of the exponential distribution
($A > 0$)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N) : Output array of dimension N containing the generated exponential variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Exponential parameter out of range
3 : Seed out of range)

RANFDI

PURPOSE

Subroutine RANFDI generates n F random variates from an F distribution with parameters IA and IB (called the numerator and denominator degrees of freedom, respectively, of the F distribution) where both IA and IB are positive integers. The form of the F distribution used is

$$f(x) = \frac{\Gamma[(IA+IB)/2]}{\Gamma(IA/2)\Gamma(IB/2)}\left(\frac{IA}{IB}\right)^{IA/2} x^{IA/2-1}\left(1+\frac{IA}{IB}x\right)^{-(IA+IB)/2} \quad \begin{array}{l} 0 < x < \infty \\ , IA = 1,2,3,\ldots \\ IB = 1,2,3,\ldots \end{array}$$

The mean and variance of the F distribution are

$$\text{mean} = IB/(IB-2) \qquad \text{for } IB > 2$$

$$\text{variance} = (2 \cdot IB^2(IA+IB-2))/(IA(IB-2)^2(IB-4)) \qquad \text{for } IB > 4$$

FEATURES

The generation scheme used in RANFDI is based on the relationship between the beta and F distributions; namely, the F with IA and IB degrees of freedom is equivalent to the following expression where Y is a beta random variable with parameters $a = IA / 2$ and $b = IB / 2$:

$$(IB \cdot Y)/(IA(1-Y))$$

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 130.
2. Krutchkoff, R. G. (1970), ***Probability and Statistical Inference***, Gordon and Breach, pp. 54 - 56.
3. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, pp. 207 - 210.

INPUT GUIDE

The call line for subroutine RANFDI is

```
CALL RANFDI (N,IA,IB,ISEED,X,IERRS)
```

or, in structured programming form,

```
     CALL RANFDI
G                ( N       , IA       , IB
B                , ISEED
Y                , X       , IERRS )
```

The parameter list is given below:

Given arguments

- N : Number of variates to be generated
  ($N > 0$)
- IA : Numerator degrees of freedom parameter for the F distribution
  ($IA > 0$)
- IB : Denominator degrees of freedom parameter for the F distribution
  ($IB > 0$)

Both

- ISEED : Integer seed
  ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

- X(N) : Output array of dimension N containing the generated F variates
- IERRS : Input error flag
  (0 : No input errors
  1 : Number of variates out of range
  2 : Numerator degrees of freedom parameter out of range
  3 : Denominator degrees of freedom parameter out of range
  4 : Seed out of range)

RANGAM

PURPOSE

Subroutine RANGAM generates n gamma random variates from a gamma distribution with parameters a and b (a and b both > 0). The form of the gamma distribution used is

$$f(x) = \frac{1}{\Gamma(a)b^a} x^{a-1} e^{-x/b} \quad \begin{matrix} 0 < x < \infty \\ , a > 0 \\ b > 0 \end{matrix}$$

The mean and variance of the gamma distribution are

mean = ab

variance = ab^2

FEATURES

The generation scheme used in RANGAM is written in two parts, keying upon whether the parameter a is integral or nonintegral. A more computationally efficient algorithm is included to handle the special case in which $a > 1$ and $b = 1$.

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 203 - 204 and pp. 208 - 209.
2. Knuth, D. E. (1981), ***The Art of Computer Programming***, Volume 2 / Seminumerical Algorithms, Second Edition, Addison-Wesley, p. 129.

INPUT GUIDE

The call line for subroutine RANGAM is

CALL RANGAM (N,A,B,ISEED,X,IERROR)

or, in structured programming form,

```
    CALL RANGAM
G                   ( N       , A       , B
B                   , ISEED
Y                   , X       , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated ($N > 0$)
A	:	Shape parameter of the gamma distribution ($A > 0$)
B	:	Scale parameter of the gamma distribution ($B > 0$)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N)	:	Output array of dimension N containing the generated gamma variates
IERROR	:	Input error flag (0 : No input errors 1 : Number of variates out of range 2 : Shape parameter A out of range 3 : Scale parameter B out of range 4 : Seed out of range)

RANLGS

PURPOSE

Subroutine RANLGS generates n logistic random variates from a logistic distribution with parameters a and b ($b > 0$). The value of a determines the location of the distribution on the abscissa. The value of b controls the degree of spread in the distribution. The form of the logistic distribution used is

$$f(x) = \left(e^{-\frac{x-a}{b}}\right) / \left(b\left(1+e^{-\frac{x-a}{b}}\right)^2\right) \quad , \quad \begin{array}{r} -\infty < x < \infty \\ -\infty < a < \infty \\ b > 0 \end{array}$$

The mean and variance of the logistic distribution are

mean = a

variance = $(b\pi)^2/3$

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 201 and p. 241.
2. Johnson, N. L. and Kotz, S. (1970), ***Continuous Univariate Distributions - 2***, Houghton Mifflin, pp. 1 - 5.

INPUT GUIDE

The call line for subroutine RANLGS is

```
CALL RANLGS (N,A,B,ISEED,X,IERROR)
```

or, in structured programming form,

```
     CALL RANLGS
G                 ( N       , A       , B
B                 , ISEED
Y                 , X       , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated (N > 0)
A	:	Location parameter (mean) of the logistic distribution
B	:	Scale parameter of the logistic distribution (B > 0)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N)	:	Output array of dimension N containing the generated logistic variates
IERROR	:	Input error flag (0 : No input errors 1 : Number of variates out of range 2 : Scale parameter B out of range 3 : Seed out of range)

RANLOG

PURPOSE

Subroutine RANLOG generates n lognormal random variates from a lognormal distribution with parameters FMU and SIG ($SIG \geq 0$). FMU is the mean and SIG is the standard deviation of the underlying normal distribution. For the normal distribution, the value of FMU determines the distribution location on the abscissa, while the value of SIG controls the degree of spread in the normal random variates. The form of the lognormal distribution used is

$$f(x) = \frac{1}{SIG \cdot x \cdot (2\pi)^{1/2}} e^{-\frac{(\ln x - FMU)^2}{2SIG^2}} \quad , \quad \begin{array}{r} 0 < x < \infty \\ -\infty < FMU < \infty \\ SIG \geq 0 \end{array}$$

The mean and variance of the lognormal distribution are

$$\text{mean} = e^{(FMU + SIG^2/2)}$$

$$\text{variance} = e^{(2FMU + SIG^2)} \cdot (e^{SIG^2} - 1)$$

FEATURES

The generation scheme used in RANLOG is based on the fact that if Y has a normal distribution with mean FMU and variance SIG^2, then $X = e^Y$ has a lognormal distribution with

$$\text{mean} \qquad FX = e^{(FMU + SIG^2/2)}$$

and

$$\text{variance} \qquad SX^2 = e^{(2FMU + SIG^2)} \cdot (e^{SIG^2} - 1)$$

The Box-Muller procedure (Reference 1) is used to generate the required normal variates.

If the user wishes to specify FX and SX^2 instead of FMU and SIG^2, the following expressions can be used to calculate the corresponding values of FMU and SIG required by subroutine RANLOG:

$$FMU = 0.5 \ln(FX^4 / (SX^2 + FX^2))$$
$$SIG^2 = \ln((SX^2 / FX^2) + 1)$$

REFERENCES

1. Box, G. E. P. and Muller, M. E. (1950), "A Note on the Generation of Random Normal Deviates", Annals of Mathematical Statistics, Volume 29, pp. 610 - 611.
2. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 214.
3. Lindgren, B. W. (1968), ***Statistical Theory***, Second Edition, The Macmillan Company, p. 176.

INPUT GUIDE

The call line for subroutine RANLOG is

```
CALL RANLOG (N,FMU,SIG,ISEED,X,IERROR)
```

or, in structured programming form,

```
      CALL RANLOG
G                 ( N       , FMU     , SIG
B                 , ISEED
Y                 , X       , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated ($N > 0$)
FMU	:	Mean of the underlying normal distribution
SIG	:	Standard deviation of the underlying normal distribution ($SIG \geq 0$)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N)	:	Output array of dimension N containing the generated log-normal variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Standard deviation of the underlying normal improperly specified
3 : Seed out of range)

RANNOR

PURPOSE

Subroutine RANNOR generates *n* normal random variates from a normal distribution with parameters *fmu* and *sig* (*sig* ≥ 0). *fmu* is the mean of the distribution and *sig* is the distribution standard deviation. The value of *fmu* determines the distribution location on the abscissa, while the value of *sig* controls the degree of spread in the normal random variates. The form of the normal distribution used is

$$f(x) = \frac{1}{sig\,(2\pi)^{1/2}} e^{-\frac{(x-fmu)^2}{2sig^2}} \quad , \quad \begin{array}{r} -\infty < x < \infty \\ -\infty < fmu < \infty \\ sig \geq 0 \end{array}$$

The mean and variance of the normal distribution are

mean = *fmu*

variance = sig^2

FEATURES

The generation scheme used in RANNOR is known as the Box-Muller procedure (see Reference 1).

REFERENCES

1. Box, G. E. P. and Muller, M. E. (1950), "A Note on the Generation of Random Normal Deviates", Annals of Mathematical Statistics, Volume 29, pp. 610 - 611.
2. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 211 - 213.

INPUT GUIDE

The call line for subroutine RANNOR is

CALL RANNOR (N,FMU,SIG,ISEED,X,IERROR)

or, in structured programming form,

```
      CALL RANNOR
G                 ( N       , FMU     , SIG
B                 , ISEED
Y                 , X       , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated ($N > 0$)
FMU	:	Mean of the normal distribution
SIG	:	Standard deviation of the normal distribution ($SIG \geq 0$)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N)	:	Output array of dimension N containing the generated normal variates
IERROR	:	Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Standard deviation of the normal improperly specified
3 : Seed out of range)

RANNVE

PURPOSE

Subroutine RANNVE generates *n* random normal vectors each of length *ip* from a multivariate normal distribution with mean vector **fmu** and variance-covariance matrix **A**. The form of the multivariate normal distribution used is

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{ip/2}|\mathbf{A}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mathbf{fmu})'\mathbf{A}^{-1}(\mathbf{x}-\mathbf{fmu})}$$

where

x	:	An *ip*-dimensional vector of random variables following a multivariate normal distribution
fmu	:	An *ip*-dimensional vector containing the means of each of the *ip* variables
A	:	A symmetric matrix of dimension *ip* by *ip* containing the variances and covariances of the *ip* variables
$\|\mathbf{A}\|$	:	The determinant of matrix **A**

Each of the *ip* variables ranges from $-\infty$ to ∞. Each element of the mean vector **fmu** ranges from $-\infty$ to ∞. The mean and variance of the multivariate normal distribution are

mean = **fmu**

variance = **A**

FEATURES

In employing RANNVE the user must choose among three input options, each pertaining to the form of the symmetric matrix **A**:

(1) **A** is input as a full variance-covariance matrix; i.e., the (i,i)th diagonal element is the variance of the ith variable while the (i,j)th off-diagonal element represents the covariance between the ith and jth variables. (Covariances can be positive, negative or zero.)

(2) **A** is input as a "pseudo-correlation" matrix in which the (i,j)th element represents the correlation (a value ranging from - 1.0 to 1.0) between the ith and jth variables and the (i,i)th diagonal element represents the standard deviation of the ith variable.

(3) **A** is input as a diagonal matrix; i.e., the (i,i)th diagonal element is the variance of the ith variable while the off-diagonal elements are all zero implying that the ip variables are uncorrelated.

The details of the generation scheme used in RANNVE are found in References 2 and 4.

REFERENCES

1. Browne, E. T. (1958), ***Introduction to the Theory of Determinants and Matrices***, University of North Carolina Press, pp. 120 - 121.
2. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 215 - 217.
3. Morrison, D. F. (1967), ***Multivariate Statistical Methods***, McGraw-Hill, pp. 80 -81.
4. Scheuer, E. and Stoller, D. S. (1962), "On the Generation of Normal Random Vectors", Technometrics, Volume 4 (May), pp. 278 - 281.

INPUT GUIDE

The call line for subroutine RANNVE is

```
CALL RANNVE (N,IP,FMU,A,IOPTION,ISEED,AC,C,Z,X,IERROR)
```

or, in structured programming form,

```
      CALL RANNVE
G                 ( N       , IP      , FMU     , A       , IOPTION
B                 , ISEED
Y                 , AC      , C       , Z       , X       , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of random variates to be generated (N > 0)
IP	:	Length of the desired random vectors (IP > 0)
FMU(IP)	:	Mean vector of the multivariate normal distribution

A(IP,IP) Input matrix for the multivariate normal distribution (**A** must be symmetric and positive definite)

IOPTION Parameter specifying which input option the user has selected for matrix **A** (IOPTION = 1, 2, or 3)

Both

ISEED : Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

AC(IP,IP) : A matrix of dimension IP by IP used in testing matrix **A** for positive definiteness

C(IP,IP) : A lower triangular matrix of dimension IP by IP used in the generation of random normal vectors

Z(IP) : A vector of dimension IP containing IP independent normal random variates each with zero mean and unit variance

X(N,IP) : Output array of dimension N by IP containing the N generated random normal vectors each of length IP

IERROR : Input error flag

(0 : No input errors
1 : Number of random normal vectors out of range
2 : Vector length out of range
3 : Matrix input option out of range
4 : At least one variance or standard deviation improperly specified
5 : Matrix **A** is not symmetric
6 : At least one correlation out of range under option 2
7 : Matrix **A** is not positive definite
8 : Seed out of range)

COMMENTS

The user should exercise care in inputting matrix **A**. Regardless of the input option chosen, **A** must be symmetric. If option 1 or 3 is selected, **A** is a variance-covariance matrix and must be positive definite. If option 2 is chosen, **A** is a pseudo-correlation matrix which is transformed within RANNVE to a variance-covariance matrix and then checked for positive definiteness. For a discussion of positive definite matrices see Reference 1.

In the user's main program the array *X(N,IP)* must be dimensioned so as to conform exactly to the input values of *N* and *IP*. For example, if the user wishes to generate 100 vectors each of length 3, the *X* array must be dimensioned as *X*(100,3) rather than to a size larger than actually required, say, *X*(500,10). Using a larger dimension than necessary will yield invalid results.

RANPDI

PURPOSE

Subroutine RANPDI is designed to generate nrn random numbers from one of the distribution types in the Pearson system of frequency curves. Any distribution which is determined by its mean (μ) and its second, third, and fourth central moments (μ_2, μ_3, μ_4) is a member of the Pearson family of distributions. This family contains distributions which are bell-shaped, J-shaped, L-shaped, and U-shaped. Within these four general shapes is a continuum of skewed, flattened, and peaked curves.

Admissible values of the moments μ, μ_2, μ_3, and μ_4 are:

$$-\infty < \mu < \infty$$

$$\mu_2 > 0$$

$$-\infty < \mu_3 < \infty$$

$$\mu_4 > 0$$

μ_2 represents the distribution variance, μ_3 is related to the degree of skewness (lack of symmetry about the mean of the distribution), and μ_4 is related to the amount of kurtosis (peakedness) that the distribution exhibits.

The Pearson family provides an excellent source of distributions which depart from normality (symmetric and bell-shaped distribution) in varying degrees of skewness and kurtosis. These latter measures of departure from normality are given by the expressions

$$\beta_1(\text{skewness}) = \mu_3^2/\mu_2^3$$

$$\beta_2(\text{kurtosis}) = \mu_4/\mu_2^2$$

For purposes of reference the values of skewness and kurtosis for the normal distribution are $\beta_1 = 0$ and $\beta_2 = 3$, respectively. Since the assumption of normality of data is integral to many statistical procedures, RANPDI can be very useful in designing simulation studies to evaluate the effect of violations of the normality assumption.

Subroutine RANPDI allows generation of random variates from nine of the distribution types in the Pearson system. Of these nine, three (Types 1, 4, and 6) are called main types while the remaining six (Types 2, 3, 5, 7, 10, and 13) are referred to as transition types.

There are limits on the admissible values of skewness and kurtosis which will give rise to one of the Pearson distribution types treated in subroutine RANPDI. Hence, not all valid choices of the moments μ, μ_2, μ_3, and μ_4 will enable the user to obtain Pearson random variates.

For further details regarding the Pearson system, especially the functional forms of the distribution types, the user should consult References 1 and 2.

FEATURES

Subroutine RANPDI takes the values of the moments μ, μ_2, μ_3, and μ_4 that the user inputs and determines the Pearson distribution type to which these moments correspond most closely. The constant term associated with this distribution is then computed, if it exists. Next, a cumulative distribution function (cdf) table of size 10000 is generated for the Pearson distribution type determined. The generation scheme used in RANPDI is based on interpolation within this cdf table.

REFERENCES

1. Elderton, W. P. and Johnson, N. L. (1969), ***Systems of Frequency Curves***, Cambridge University Press, pp. 35 - 95.
2. Taub, A. E. (1974), ***DURG - A Documentation of the Dahlgren Universal Random Number Generator***, NWL TN-K-17/74, NSWC, Dahlgren, VIRGINIA 22448.

INPUT GUIDE

The call line for subroutine RANPDI is

```
CALL RANPDI (NRN,MU,ISEED,PDRN,IDIST,IERROR)
```

or, in structured programming form,

```
    CALL RANPDI
G                   ( NRN     , MU
B                   , ISEED
Y                   , PDRN    , IDIST     , IERROR )
```

The parameter list is given below:

Given arguments

NRN : Number of variates to be generated
(NRN > 0)

MU(4) : Input array of dimension 4 containing the moments μ, μ_2, μ_3, and μ_4 of the distribution from which variates are to be generated
(MU(2) > 0 and MU(4) > 0)

Both

ISEED : Integer seed
$(1 \leq ISEED < 2^{31} - 1)$

Yielded arguments

PDRN (NRN) : Output array of dimension NRN containing the generated Pearson variates

IDIST : Pearson distribution type determined by the values of the moments MU(1), MU(2), MU(3), AND MU(4)

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : MU(2) and/or MU(4) out of range
3 : Seed out of range
4 : The moments MU(1), MU(2), MU(3), AND MU(4) do not determine any of the Pearson distribution types treated by subroutine RANPDI
5 : Constant term for the Pearson distribution type selected on the basis of the moments MU(1), MU(2), MU(3), AND MU(4) cannot be computed)

COMMENTS

A graph displaying the admissible region of values for skewness and kurtosis can be found on page 3 of Reference 2. This can aid the user in determining input values for the moments μ, μ_2, μ_3, and μ_4. If the skewness and kurtosis values do not lie in the admissibility region, subroutine RANPDI will return IERROR = 4.

RANTDI

PURPOSE

Subroutine RANTDI generates *n T* random variates from a T distribution with parameter *NU* (called the degrees of freedom of the T distribution) where *NU* is a positive integer. The form of the T distribution used is

$$f(x) = \frac{\Gamma[(NU+1)/2]}{\Gamma(NU/2)(\pi \cdot NU)^{1/2}} \left(1 + \frac{x^2}{NU}\right)^{-(NU+1)/2} \quad , \quad \begin{array}{l} -\infty < x < \infty \\ NU = 1, 2, 3, \ldots \end{array}$$

The mean and variance of the T distribution are

mean = 0	for $NU > 1$
variance = NU / (NU - 2)	for $NU > 2$

FEATURES

The generation scheme used in RANTDI is based on the relationship between the normal, chi-square and T distributions; namely, the T distribution with *NU* degrees of freedom is equivalent to the ratio of a standard normal variate (*Z*) to the square root of a chi-square variate (*Y*) with *NU* degrees of freedom divided by its degrees of freedom:

$$Z / (Y / NU)^{1/2}$$

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 213.
2. Krutchkoff, R. G. (1970), ***Probability and Statistical Inference***, Gordon and Breach, pp. 56 - 57.
3. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, pp. 202 - 207.

INPUT GUIDE

The call line for subroutine RANTDI is

CALL RANTDI (N,NU,ISEED,X,IERRS)

or, in structured programming form,

```
    CALL RANTDI
G               ( N       , NU
B               , ISEED
Y               , X       , IERRS )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated
($N > 0$)

NU : Degrees of freedom parameter of the T distribution
($NU > 0$)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N) : Output array of dimension N containing the generated T variates

IERRS : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Degrees of freedom parameter out of range
3 : Seed out of range)

RANUNI

PURPOSE

Subroutine RANUNI generates n uniform random variates from a continuous uniform distribution over the interval a to b $(a < b)$. The form of the continuous uniform distribution used is

$$f(x) = 1/(b-a) \quad , \quad \begin{matrix} a < x < b \\ -\infty < a < b < \infty \end{matrix}$$

The mean and variance of the continuous uniform distribution are

$$\text{mean} = (a+b)/2$$

$$\text{variance} = (b-a)^2/12$$

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 200 - 202.

INPUT GUIDE

The call line for subroutine RANUNI is

```
CALL RANUNI (N,A,B,ISEED,X,IERROR)
```

or, in structured programming form,

```
      CALL RANUNI
G                 ( N       , A       , B
B                 , ISEED
Y                 , X       , IERROR )
```

The parameter list is given below:

Given arguments

N : Number of variates to be generated (N > 0)

A : Lower limit of the generation interval
(A < B)

B : Upper limit of the generation interval
(B > A)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N) : Output array of dimension N containing the generated uniform variates

IERROR : Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Generation interval improperly specified
3 : Seed out of range)

RANCIR

PURPOSE

Subroutine RANCIR generates n pairs of points uniformly within a circle of radius a centered at the point (h_x, h_y). At the user's option, the points are generated in either rectangular (i.e., cartesian) or polar coordinates. If polar coordinates are requested, the generated angles are positive and are expressed in degrees.

RANCIR utilizes the following sets of equations relating rectangular (x, y) and polar (r, θ) coordinates (see, for example, Reference 2):

$$\begin{aligned} x &= r\cos\theta \\ y &= r\sin\theta \end{aligned} \tag{1}$$

and

$$\begin{aligned} r^2 &= x^2 + y^2 \\ \tan\theta &= y/x \end{aligned} \tag{2}$$

The set (1) transforms polar coordinates to rectangular coordinates, while the set (2) performs the reverse transformation.

REFERENCES

1. Crigler, J. R., "A Note on the Generation of Coordinate Pairs Uniformly Within a Circle of Radius A", unpublished note to K106 files dated April 9, 1982.
2. Goodman, A. W. (1965), ***Analytic Geometry and the Calculus***, The Macmillan Company, pp. 375 - 377.

INPUT GUIDE

The call line for subroutine RANCIR is

CALL RANCIR (N,A,HX,HY,IT,ISEED,X,Y,IERROR)

or, in structured programming form,

```
      CALL RANCIR
G                 ( N       , A       , HX      , HY      , IT
B                 , ISEED
Y                 , X       , Y       , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of pairs to be generated (N > 0)
A	:	Radius of the circle (A > 0)
HX	:	X-coordinate (cartesian) of the center of the circle
HY	:	Y-coordinate (cartesian) of the center of the circle
IT	:	Type of coordinates requested, i.e., = 1 for cartesian coordinates = 2 for polar coordinates

Both

ISEED	:	Integer seed $(1 \leq ISEED < 2^{31} - 1)$

Yielded arguments

X(N)	:	Output array of dimension N containing the generated x-coordinates (radial distances if polar coordinates requested)
Y(N)	:	Output array of dimension N containing the generated y-coordinates (positive angles in degrees if polar coordinates requested)
IERROR	:	Input error flag (0 : No input errors 1 : Number of pairs out of range 2 : Circle radius out of range 3 : Coordinate type out of range 4 : Seed out of range)

COMMENTS

The user should note that the circle center must be specified in cartesian coordinates on input.

RANWEI

PURPOSE

Subroutine RANWEI generates n Weibull random variates from a Weibull distribution with parameters a, b, and c. a is interpreted as the location parameter (determines where the distribution is on the abscissa), b as the shape parameter ($b > 0$), and c as the scale parameter ($c > 0$) of the Weibull. The form of the Weibull distribution used is

$$f(x) = \frac{b}{c^b}(x-a)^{b-1}e^{-\left(\frac{x-a}{c}\right)^b} \quad , \quad \begin{matrix} a < x < \infty \\ -\infty < a < \infty \end{matrix} \quad , \quad \begin{matrix} b > 0 \\ c > 0 \end{matrix}$$

The mean and variance of the Weibull distribution are

$$\text{mean} = a + c \cdot \Gamma\left(\frac{b+1}{b}\right)$$

$$\text{variance} = c^2\left(\Gamma\left(\frac{b+2}{b}\right) - \left[\Gamma\left(\frac{b+1}{b}\right)\right]^2\right)$$

REFERENCES

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., p. 211.
2. Johnson, N. L. and Kotz, S. (1970), ***Continuous Univariate Distributions - 1***, Houghton Mifflin, pp. 250 - 252.
3. Walpole, R. E. and Myers, R. H. (1985), ***Probability and Statistics for Engineers and Scientists***, Third Edition, Macmillan, pp. 156 - 157.

INPUT GUIDE

The call line for subroutine RANWEI is

CALL RANWEI (N,A,B,C,ISEED,X,IERROR)

or, in structured programming form,

```
    CALL RANWEI
G                ( N      , A      , B      , C
B                , ISEED
Y                , X      , IERROR )
```

The parameter list is given below:

Given arguments

N	:	Number of variates to be generated ($N > 0$)
A	:	Location parameter of the Weibull
B	:	Shape parameter of the Weibull ($B > 0$)
C	:	Scale parameter of the Weibull ($C > 0$)

Both

ISEED	:	Integer seed ($1 \leq ISEED < 2^{31} - 1$)

Yielded arguments

X(N)	:	Output array of dimension N containing the generated Weibull variates
IERROR	:	Input error flag

(0 : No input errors
1 : Number of variates out of range
2 : Shape parameter out of range
3 : Scale parameter out of range
4 : Seed out of range)

RANMK1

PURPOSE

Subroutine RANMK1 generates an autocorrelated sequence, X_n, of length n where X_n is a first-order Markov process with parameter *alpha* (-1 < *alpha* < 1). The sequence is generated via the recursive expression

$$X_n - fmu = alpha \cdot (X_{n-1} - fmu) + Z_n .$$

Z_n is the random error term, assumed to be normally distributed with mean 0 and standard deviation *sig* (*sig* > 0). The generation scheme assumes that X_n is a normal process with mean *fmu* and variance given by

$$sig^2 / (1 - alpha^2) .$$

X_n will be a stationary process provided that -1 < *alpha* < 1. We note that a first-order Markov process is synonymous with a first-order autoregressive (AR) process.

REFERENCE

1. Fishman, G. S. (1973), ***Concepts and Methods in Discrete Event Digital Simulation***, John Wiley & Sons, Inc., pp. 236 - 237.

INPUT GUIDE

The call line for subroutine RANMK1 is

```
CALL RANMK1 (N,FMU,SIG,ALPHA,ISEED,Y,X,IERROR)
```

or, in structured programming form,

```
      CALL RANMKI
G                 ( N       , FMU     , SIG       , ALPHA
B                 , ISEED   , Y
Y                 , X       , IERROR )
```

The parameter list is given below:

Given arguments

N : Length of the sequence to be generated
(N > 0)

FMU : Mean of the normal process

SIG : Standard deviation of the normally distributed error term
(SIG > 0)

ALPHA : Value of the Markov model parameter
(-1 < ALPHA < 1)

Both

ISEED : Integer seed
($1 \leq ISEED < 2^{31} - 1$)

Y(N) : Array of dimension N containing the normally distributed error terms for the first-order Markov process

Yielded arguments

X(N) : Output array of dimension N containing the generated first-order Markov sequence

IERROR : Input error flag

(0 : No input errors
1 : Sequence length out of range
2 : Standard deviation of the normally distributed error term improperly specified
3 : First-order Markov model parameter improperly specified
4 : Seed out of range)

GLOSSARY

The documentation of the programs and subroutines in STATLIB contain several instances of mathematical notation which may not be familiar to the reader. Rather than repeatedly explaining this notation each time it appears, these items have been collected into this glossary for definition and easy reference.

1. $n!$ Factorial notation;

$$n! = 1 \times 2 \times 3 \times \ldots \times (n-1) \times n.$$

2. $\binom{n}{x}$ Combinatorial notation; the number of combinations of n different things taken x at a time;

$$\binom{n}{x} = \frac{n!}{x!(n-1)!}.$$

3. $\Gamma(x)$ Gamma function;

$$\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt, \quad 0 < x < \infty.$$

4. $I_x(a,b)$ Incomplete Beta function ratio;

$$I_x(a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^x t^{a-1}(1-t)^{b-1} dt \qquad \begin{matrix} 0 \le x \le 1 \\ a,b > 0 \end{matrix}$$

DISTRIBUTION

Commander
Naval Air Development Center
Attn: Technical Library
Warminster, PA 18974
1

President
Naval War College
Newport, RI 02840
1

Superintendent
U. S. Naval Academy
Attn: Mathematics Department
Annapolis, MD 21402
1

Director
David Taylor Research and
Development Center
Attn: Technical Library
Bethesda, MD 20084
1

Director
Naval Research Laboratory
Attn: Technical Information
Division (2600)
4555 Overlook Ave.
S. W. Code 2000
Washingtion, D. C. 20375
1

Office of the Chief of Naval Reasearch
Department of the Navy
Attn: Director,
Mathematical Sciences Division
800 N. Quincy St.
Arlington, VA 22217
1

Center for Naval Analyses
4401 Ford Ave.
P. O. box 16268
Arlington, VA 22302
1

Superintendent
U. S. Naval Postgraduate School
Attn: Technical Library
Monterey, CA 93940
1

Commanding Officer
Naval Weapons Center
Attn: Technical Library
China Lake, CA 93555
1

Commander
U. S. Army Missile Research,
Development, and Engineering
Center
Redstone Arsenal, AL 35898
1

Commanding Officer
ARDEC
Picatinny Arsenal
Picatinny Arsenal, NJ 07806
1

Director
U. S. Army Ballistic Research Laboratory
Attn: Technical Library
Aberdeen Proving Ground, MD 21005
1

Commanding Officer
U. S. Army CAA
Attn: Carl Bates
8120 Woodmont Avenue
Bethesda, MD 20814
1

Commanding Officer
U. S. Army Research Office
Attn: Dr. Francis Dressel
P. O. Box 12211
Research Triangle Park, NC 27709
1

Director
Harry Diamond Laboratories
2800 Powder Mill Road
Adelphi, MD 20783
1

FMC
Bayberry Office Park
Dahlgren, VA 22448 — 1

Business and Technological Systems, Inc.
Attn: Technical Library
Aerospace Bldg., Suite 440
Seabrook, MD 20706 — 1

EG&G Washington Analytical Services Center, Inc.
Attn: Technical Library
P. O. Box 552
Dahlgren, VA 22448 — 1

Los Alamos National Laboratory
P. O. Box 1663
Attn: Library
Los Alamos, NM 87545 — 1

U. S. Army Mathematics Research Center
University of Wisconsin
Attn: Technical Library
Madison, WI 54306 — 1

Commanding Officer
U. S. Air Force Armament Laboratory
Eglin Air Force Base, FL 32542 — 1

ASG Corporation
Attn: Jack Warner
Dahlgren, VA 22448 — 1

Department of Statistics
VPI&SU
Attn: Dr. Klaus Hinkelmann
Blacksburg, VA 24060 — 1

Department of Computer Science
VPI&SU
Attn: Dr. Donald C. S. Allison
Blacksburg, VA 24060 — 1

Department of Statistics
North Carolina State University
Raleigh, NC 27105 — 1

The Johns Hopkins University
Applied Physics Laboratory
Attn: Document Library
Johns Hopkins Road
Laurel, MD 20707 — 1

Applied Physics Laboratory
The Johns Hopkins University
8621 Georgia Avenue
Attn: Document Library
Silver Spring, MD 20910 — 1

Oklahoma State University Field Office
P. O. Box 1925
Attn: Technical Library
Eglin Air Force Base, FL 32542 — 1

Department of Statistics
University of Iowa
Iowa City, IA 50703 — 1

Department of Statistics
Iowa State University
Ames, IA 50010 — 1

Virginia Commonwealth University
Biometry Department
Attn: Dr. Hans Carter
Richmond, VA 23219 — 1

Department of Statistics
Purdue University
Lafayette, IN 47905 — 1

The Rand Corporation
1600 Main Street
Santa Monica, CA 90406 — 1

Naval Coastal Systems Center
Attn: Technical Library
Panama City, FL 32407 — 1

Aerojet ElectroSystems 1
P. O. Box 296
Attn Maryann Becker 160/4343
Azusa, CA 91702

Control Data Corporation 1
Attn: William E. Glass
8100 34th Avenue, South
Bloomington, MN 55431

Sandia National Laboratories 1
P. O. Box 5800
Albuquerque, NM 87185

Mary Washington College 1
Mathematics Department
Attn: Dr. Manning G. Collier
Fredericksburg, VA 22401

ClearVu Technical Writing Services 1
Attn: Thomas Dunbar
95 Clearview Drive
Christiansburg, VA 24073

Ms. Audrey E. Taub 1
9 Foster Rd.
Burlington, MA 01803

Ms. Patricia A. Shields 1
DCSC-LO
P. O. Box 3990
Columbus, Ohio 43216-5000

Library of Congress 4
Attn: Gift and Exchange Division
Washington, D. C. 20540

Internal Distribution:

C	1	E43	1	G11 (W. Moore)	1
D	1	E43 (A. Morris)	1	G12 (W. Chow)	1
D1	1	E50	1	G20	1
D2	1	F	1	G30	1
D21	1	F06 (Dr. H. Crisp)	1	G32 (B. Jablovskis)	1
D25	1	F10	1	G32 (C. Smith)	1
E	1	F20	1	G40	1
E02	1	F22 (C. Frame)	1	G60	1
E20	1	F30	1	G70	1
E211	1	F40	1	H	1
E231	10	G	1	H10	1
E232	1	G04 (Dr. W. Soper)	1	H20	1
E30	1	G10	1	H30	1
E31 (GIDEP)	1	G102 (G. Hombaker)	1	K	1
E40	1	G11 (D. Clawson)	1	K02	1
E406	200	G11 (Dr. T. Goswick)	1	K10	1

K104 (Dr. A. DiDonato)	1	K43 (M. Wells)	1	R	1
K20	1	K44	1	R10	1
K40	1	K44 (M. Busby)	1	R30	1
K404 (T. Alexander)	1	K44 (M. Gallier)	1	R33 (Dr. C. Dacres)	1
K41	1	K50	1	R40	1
K42	1	K52	1	U	1
K42 (Dr. W. Farr)	1	N	1	U10	1
K43	1	N10	1	U20	1
K43 (R. Fondren)	1	N20	1	U30	1
K43 (Dr. E. Maddry)	1	N30	1	U40	1
K43 (G. Sitzman)	1	N40	1	X	1

REPORT DOCUMENTATION PAGE

1 OFFICIAL USE ONLY	2 REPORT DATE	3 TYPE OF REPORT AND DATES COVERED
	August 1989	Final

4 TITLE AND SUBTITLE

STATLIB: NSWC LIBRARY OF STATISTICAL PROGRAMS AND SUBROUTINES

5 FUNDING NUMBERS

6 AUTHOR(S)

Marlin A. Thomas, Gary W. Gemmill, and John R. Crigler

7 PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Surface Warfare Center
Code E406
Dahlgren, VA 22448-5000

8 PERFORMING ORGANIZATION REPORT NUMBER

NSWC TR 89-97

9 SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10 SPONSORING/MONITORING AGENCY REPORT NUMBER

11 SUPPLEMENTARY NOTES

12a DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b DISTRIBUTION CODE

13 ABSTRACT (MAXIMUM 200 WORDS)

This document provides a description of each program and subroutine in STATLIB, THE Naval Surface Warfare Center library of statistical programs and subroutines for its general purpose computers. The Library contains thirty-four programs and twenty-four subroutines for statistical analysis, probability evaluation, and random number generation. It was written to enable Center Scientists and Engineers to efficiently perform a wide variety of analyses and to generate pseudo random numbers from many different probability distributions.

14 SUBJECT TERMS

15 NUMBER OF PAGES
287

16 PRICE CODE

17a SECURITY CLASSIFICATION (OF REPORT)	17b SECURITY CLASSIFICATION (OF PAGE)	17c SECURITY CLASSIFICATION (OF ABSTRACT)	18 MEDIA TYPE
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	

INSTRUCTIONS FOR COMPLETING THE REPORT DOCUMENTATION PAGE

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistant with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to stay within the lines to meet optical scanning requirements.

Block 1. Official Use Only. Leave blank. Used for Government Accession Number or recipient's catalog number, (*eg.* AD A159 131, DE85010036, N86-34124, PB86-106184).

Block 2. Report Date. Full publication date including day, month, and year if available. Must cite at least the year.

Block 3. Type of Report and Dates Covered. State interim, final, etc., and, if applicable, inclusive dates, (*eg.* 10 Jun 84 - 30 Jun 85).

Block 4. Title and Subtitle. A title is taken from the part of report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C = Contract	PR = Project
G = Grant	TA = Task
PE = Program Element	WU = Work Unit

Block 6. Author. Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler this should follow the name(s).

Block 7. Performing Organization Name and Address. Give name, city, state, and zip code and telephone number if known.

Block 8. Performing Organization Report Number.

Block 9. Sponsoring/Monitoring Agency Name and Address. Give name, city, state, and zip code.

Block 10. Sponsoring/Monitoring Agency Report Number. (If known).

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with... Trans. of...To be published in...When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution Statement. Denotes public availability or limitations. Cite any availability to the public, with DTIC (*see DoDD 5230.24, Distribution Statements on Technical Documents.*) DOE (*see authorities*) NASA (*see Handbook NHB 2200.2*) NTIS (*leave blank*). Enter additional limitations or special markings in all capitols, *eg.* NOFORN, REL, ITAR.

Block 12b. Distribution Code.
DTIC (*leave blank*)
DOE (*enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports*)
NASA (*leave blank*)
NTIS (*leave blank*)

Block 13. Abstract. Include a brief (200 words or less) factual summary of the most siginificant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code. *(NTIS only).*

Block 17a. Security Classification of This Report. Enter U. S. Security Classification in accordance with U. S. Security Regulations (*i.e.*, UNCLASSIFIED).

Block 17b. Security Classification of This Form. Enter U. S. Security Classification in accordance with U. S. Security Regulations (*i.e.*, UNCLASSIFIED).

Block 17c. Security Classification of This Abstract. Enter U. S. Security Classification in accordance with U. S. Security Regulations (*i.e.*, UNCLASSIFIED).

Block 18. Media Type. Enter media type, *eg.* disc, video, AV, fiche.