5-1969

Computer Programs for Incomplete Block Designs

Fred Miller
Utah State University

COMPUTER PROGRAMS FOR INCOMPLETE BLOCK DESIGNS

by

Fred K. Miller

A report submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Applied Statistics

Plan B

Approved:

UTAH STATE UNIVERSITY

Logan, Utah

1969

TABLE OF CONTENTS

INTRODUCTION

In most disciplines where research is involved, there exists an occasional problem of having minimal facilities and/or funds for conducting experiments. This often necessitates the use of designs known as incomplete block designs.

Since the calculations needed to provide an appropriate statistical analysis are somewhat tedious, particularly in the larger designs, it is advantageous to have computer programs to do the necessary calculations.

There are several computer programs at Utah State University written in Fortran II language with Forcom subroutines that perform the analyses for incomplete block designs. These programs, for the most part, were authored by Justus Seely and Dr. Rex L. Hurst, Head, Department of Applied Statistics and Computer Science at U.S.U.

Only four of these available programs are covered in this paper. These programs are converted to Fortran IV language using the binary coded decimal system (BCD). The programs involved are Balanced Lattices, Designs Arranged in Replications, Designs Arranged in Groups of Replications, and Designs Not Arranged in Replications or Groups of Replications. The three latter programs will hereinafter be referred to as Type I, Type II, and Type III, respectively.

The programs will be written in double precision instead of single precision in order to increase computational accuracy. In single precision the number of significant digits carried accurately in the IBM 360 mod 44 computer is limited to seven. Since the numbers that represent the data need to be squared and summed, the size of number that could be used would have to be rather small, preferably less than four digits. With double precision the number of digits carried accurately is fourteen. This now presents the opportunity to use larger numbers to represent the data, but it is advisable to use numbers with less than six digits if possible.

BALANCED LATTICE
(BALLAT)

I. DESCRIPTION:

This program will compute the analysis of variance, the unadjusted and adjusted treatment means, and pertinent statistics. The program is presently set up to handle up to 81 treatments, but this can easily be changed by making the appropriate modifications in the dimension statement.

There is an option in the program to have the adjusted and unadjusted treatment means punched on cards as well as printed at the user's discretion. A basic knowledge of Fortran is advisable but not necessary.

II. METHODOLOGY:

The data cards are read in by use of a DØ loop reading the block number first and the data second. The individual replication, block, and treatment totals are computed during the course of the DØ loop that is reading the data. The appropriate sums of squares are then calculated using the procedures described by Cochran and Cox, "Experimental Designs," pp. 396-403.

III. INPUT:

The input consists of three types of cards preceding the data cards. The first card is the control card which defines the number of treatments and the number of replications in the experiment. The format card describing how the data cards are to be read comes next.

The last card is the sequence designation card.

CONTROL CARD
(4I5, 20X, 10A4)

Columns		Description
(4- 5)	9-81	Number of treatments
(9-10)	4-10	Number of replications
(15)		Type of output wanted 0 = Print unadjusted and adjusted treatment means 1 = Print and punch unadjusted, adjusted treatment means, and standard error of the difference between two treatment means
(20)		Data input device code 1-4 = Tape, disc, etc. (see COMMENTS) 5 = Card reader
(41-80)		Descriptive information

The number of treatments must be an exact square (k^2). The number of replications is equal to (k + 1).

INPUT FORMAT SPECIFICATION

The first column must be a left parenthesis (punched on either an 026 or 029 key punch). This is followed by the Fortran format of the data cards. This consists of any combination of wX and Fw.d specifications. The Fw.d is used to read both the block number and the desired variable. Up to 80 columns may be used for the format expression.

SEQUENCE DESIGNATION CARD
(2I3)

Columns	Description
(1- 3)	Position of block number in read list
(4- 6)	Position of data in read list

There are times when the block number will not be punched in columns preceding the data. To make allowances for this, a sequence designation card is required to rearrange the input sequence. As an example, if the data card format were:

Columns	Description
(1- 7)	Not used
(8-11)	Y_1 (XX.XX)
(12-15)	Not used
(16-17)	Block number

the format statement would be:

```
(7X, F4.2, 4X, F2.0)
       ↓        ↓
     Data    Block No.
```

This means that the data field is being read in first and the block number is being read second. Therefore, the proper sequence designation would be 2,1. This means that the second element being read is the block number and the first element is the data.

DATA CARDS

The data cards must be punched so that they can be read with a Fortran read statement. No alpha-numeric codes may be used for Block identification (block numbers). The block numbers must be consecutive integers starting with one. The data fields can be punched either with or without decimals, but the procedure must remain constant once the decision is made. If negative data is required, then an extra column must be allowed for the negative sign. The negative sign is made by using an eleven punch to the left of the most significant digit in the field. The data cards must be sorted, treatments within reps. If this is not done, the adjusted treatment means and the calculated F-test will not be correct.

IV. OUTPUT:

All cards preceding the data are read in and printed out. The next thing printed out is the completed analysis of variance. The rest of the output is extensively labeled and no explanation will be given here.

V. COMMENTS:

Even though it is not mandatory to punch the treatment number and replication number in the cards, it is more convenient to have them there for sorting purposes.

The program is set up to read only one variable at a time. If the user wishes to analyze multiple variables, then he will first need to put the data on either disk or tape by using some of the available utility programs or writing his own program to accomplish this. After the data has been put on tape or disk, the variables can then be obtained individually by using the appropriate format statement.

Batch processing is assumed.

The dimension may be changed if a larger number of treatments is desired. The dimension for replications should also be modified if the dimension for treatments is changed.

If the option is used to both print and punch the unadjusted, adjusted treatment means, and the standard errors of the difference of two means, the format and sequence for the cards is the same as for what is printed. The first card punched contains the unadjusted and adjusted standard errors of the difference between two means in that order. On the following cards the treatment number is punched first, then the unadjusted treatment mean is punched second and the adjusted treatment mean is punched last. The decimal is punched in the output cards.

VI. SAMPLE PROBLEM:

The following example is taken from Cochran and Cox, "Experimental Designs," pp. 396-400. There were 9 treatments and 4 replications with the data cards punched in the following format:

Columns	Description
(1- 2)	Block number
(3- 5)	Not used (trt. no. and rep. no. were punched here)
(6- 8)	Data (X.XX)

The DØ loop reads in the data cards and some basic calculations are performed within the loop as follows:

Trt.	Total
1	6.97
2	7.36
3	8.05
4	6.42
5	4.01
6	7.62
7	5.46
8	5.61
9	5.92

Block	Total
1	6.22
2	4.76
3	4.09
4	3.54
5	4.78
6	5.78
7	4.08
8	5.72
9	4.41
10	4.77
11	4.52
12	4.75

Rep	Total
1	15.07
2	14.10
3	14.21
4	14.04

The following analysis of variance is derived from these and later calculations:

Source	d.f.	Sums of Squares	Mean Squares
Total	35	5.9609	
Replication	3	0.0774	
Treatments	8	3.2261	
Blocks (Adj.)	8	1.4206	0.1776
Error	16	1.2368	0.0773

The reader is reminded that it is invalid to make an F-test with the above information. The adjusted F ratio that is calculated and printed out last is computed in the following manner. The sum of squares of deviations of the adjusted treatment totals is calculated first. This is then divided by the number of replications and the quotient is divided by (k^2-1) where k^2 represents the number of treatments. This final result is divided by the effective error mean square. The error mean square from the analysis of variance table above is multiplied by the quantity $(1 + k\mu)$, where $k = \sqrt{k^2}$ and $\mu = \frac{(E_b - E_e)}{k^2 E_b}$ to obtain the effective error mean square. The E_b and E_e used in the last equation refer to the blocks (adj.) mean square and error mean square, respectively.

DESIGNS ARRANGED IN REPLICATIONS (TYPE I)

I. DESCRIPTION:

This program was written to serve as a utility program in a statistical laboratory. It will compute the analysis of variance, the unadjusted and adjusted treatment means, and other pertinent statistics. At the present time this program can handle up to 100 treatments. If more treatments are desired, then a simple modification of the dimension statement is all that is necessary.

At the user's discretion the program can punch the unadjusted and adjusted treatment means on cards in addition to having these means printed. It is advisable to have some knowledge of Fortran language.

II. METHODOLOGY:

A DØ loop reads the data cards. The treatment number is read first and the data field second. Some of the calculations performed during this loop are total sums of squares (uncorrected), block sums of squares, block and replication totals, and the grand total of all observations. The other necessary computations are then performed using the procedures described by Cochran and Cox, "Experimental Designs," pp. 443-446.

III. INPUT:

The input consists of three types of cards preceding the data cards. The first card, called the control card, defines the number of treatments, number of units per block, number of replications, and number of blocks within each replication in the experiment. The second card is the format card which describes how the data cards are to be read. The last card is the sequence designation card.

CONTROL CARD
(6I5, 10X, 10A4)

Columns		Description
(3- 5)	4-100	Number of treatments
(9-10)	2- 10	Number of units per block
(14-15)	3- 10	Number of replications
(18-20)		Number of blocks within each replication
(25)		Type of output wanted 0 = Print unadjusted and adjusted treatment means 1 = Print and punch unadjusted, adjusted treatment means, and standard error of the difference between two treatment means
(30)		Data input device code 1-4 = Tape, disk, etc. (see COMMENTS) 5 = Card reader
(41-80)		Descriptive information

INPUT FORMAT SPECIFICATION

The first column must be a left parenthesis (punched on either an 026 or 029 key punch). This is followed by the Fortran format of the data cards. This consists of any appropriate combination of wX and Fw.d specifications. The Fw.d is used to read in both the treatment number and the desired variable. Up to 80 columns may be used for the format expression.

SEQUENCE DESIGNATION CARD (2013)

Columns	Description
(2- 3)	Position of treatment number in read list
(5- 6)	Position of data field in read list

There are times when the treatment number will not be punched in columns preceding the data. To make allowances for this, a sequence designation card is required to rearrange the input sequence. As an example, if the data card format were:

Columns	Description
(1- 7)	Not used
(8-11)	Y_1 (XX.XX)
(12-15)	Not used
(16-17)	Treatment number

the format statement would be:

```
(7X, F4.2, 4X, F2.0)
      ↓          ↓
     Data    Treatment number
```

This means that the data is being read in first and the treatment number is being read second. Therefore, the proper sequence designation would be 2,1. This means that the second element being read is the treatment number and the first element is the data.

DATA CARDS

The data cards must be punched so that they can be read with a Fortran read statement. No alpha-numeric codes may be used for treatment numbers. The treatment numbers must be consecutive integers starting with one. The data fields may or may not be punched with a decimal, but the procedure must remain constant for all data cards once the decision is reached. If negative data is required, then an extra column must be allowed for the negative sign. The negative sign is made by using an eleven punch to the left of the most significant digit in the field. The data cards must be sorted blocks within replications. An invalid F-test and erroneous adjusted treatment means will result if this is not done.

IV. OUTPUT:

The cards preceding the data cards are first read and then printed out. The next thing printed out is the completed analysis of variance table. The rest of the output is extensively labeled and no explanation will be given here.

V. COMMENTS:

Even though it is not a requirement to the program to have the replication and block numbers punched on each card, it would be advisable to do so for ease in sorting the cards in the previously specified manner.

The program is presently set up to read only one data field at a time. If the user wishes to analyze multiple data fields, then he will first need to put all of the data on either disk or tape by using some of the available utility programs or creating his own program to accomplish this. After the data has been put on disk or tape, the data fields can then be obtained individually by using the appropriate format statement. Batch processing is assumed.

The dimension may be changed if a larger number of treatments, units per block, and/or replications are desired.

If the option is used to both print and punch the unadjusted, adjusted treatment means, and the standard errors of the difference of two means, the format and sequence for the cards is the same as for what is printed. The first card punched contains the unadjusted and adjusted standard errors of the difference between two means in that order. On the following cards the treatment number is punched first, the unadjusted treatment mean is punched second, and the adjusted treatment mean is punched last. The decimal is also punched in the output cards.

VI. SAMPLE PROBLEM:

The following problem was taken from Cochran and Cox, "Experimental Designs," pp. 443-446. There were 6 treatments, 2 units per block, 5 replications, and 3 blocks within each replication for this problem. The data cards were punched in the following manner:

Columns	Description
(1)	Not used
(2)	Treatment number

(3)	Not used
(4- 5)	Data (XX.)

The DØ loop reads in the data cards and some basic calculations are performed within the loop as follows:

Block	Total	Block	Total	Block	Total	Rep	Total
1	24	6	59	11	59	1	137
2	51	7	35	12	66	2	153
3	62	8	63	13	38	3	148
4	44	9	50	14	45	4	190
5	50	10	65	15	58	5	141

Grand total = 769

The following analysis of variance is then derived:

Source	d.f.	Sums of Squares	Mean Squares
Total	29	1649.0	
Replication	4	298.5	
Unadjusted treatments	5	1059.8	
Blocks within replications	10	213.4	21.34
Error	10	77.3	7.73

It is invalid to make an F-test in the conventional manner using the above information. The adjusted F-ratio that is calculated and printed out last is computed in the following manner. The sum of squares of deviations of the adjusted treatment totals is calculated first. This is then divided by r(t-1) where r represents the number of replications and t refers to the number of treatments. The quotient is then divided by the effective error mean square to give an F-test. The effective error mean square is computed using the formula $E_e \left[1+(t-k)\mu \right] = E_e'$, where E_e is error mean square, and k is the number of units per block from the analysis of variance table of above. The weighting factor, μ, is interpreted as

$$\mu = \frac{r(E_b-E_e)}{rt(k-1)E_b+k(b-r-t+1)E_e}$$

where all symbols are the same as previously defined, b is the total number of blocks, and E_b is the blocks within replications (adj.).

DESIGNS ARRANGED IN GROUPS OF REPLICATIONS (TYPE II)

I. DESCRIPTION:

This program will compute the analysis of variance, the unadjusted and adjusted treatment means, and other pertinent statistics. At the present time, this program is set up to handle 100 treatments. If more treatments are needed, then a simple modification of the dimension statement is all that is necessary.

At the user's discretion, the program can punch the unadjusted and adjusted treatment means on cards in addition to having these means printed. It is advisable to have some knowledge of Fortran language.

II. METHODOLOGY:

A DØ loop reads the data cards. The treatment number is read first and the data field second. Some of the calculations performed during this loop are total sums of squares (unadjusted), block sums of squares, block and replication totals, and the grand total of all observations. The other necessary computations are then performed using the procedures described by Cochran and Cox, "Experimental Designs," p. 446.

III. INPUT:

The input consists of three types of cards preceding the data cards. The first card, called the control card, defines the number of treatments, number of units per block, number of replications, number of blocks within each replication, and number of groups. The second card is the format card which describes how the data cards are to be read. The last card is the sequence designation card.

CONTROL CARD
(7I5, 5X, 10A4)

Columns		Description
(3- 5)	4-100	Number of treatments
(9-10)		Number of units per block
(14-15)		Number of replications
(19-20)		Number of blocks within each group
(24-25)		Number of groups
(30)		Type of output wanted 0 = Print unadjusted and adjusted treatment means 1 = Print and punch unadjusted, adjusted treatment means, and the standard error of the difference between two treatment means
(35)		Data input device code 1-4 = Tape, disk, etc. (see COMMENTS) 5 = Card reader
(41-80)		Descriptive information

INPUT FORMAT SPECIFICATION

The first column must be a left parenthesis (punched on either an 026 or 029 key punch). This is followed by the Fortran format of the data cards. This consists of any appropriate combination of wX and Fw.d specifications. The Fw.d is used to read both the treatment number and the desired variable. Up to 80 columns may be used for the format expression.

SEQUENCE DESIGNATION CARD
(2013)

Columns	Description
(2- 3)	Position of treatment number in read list
(5- 6)	Position of data field in read list

There are times when the treatment number will not be punched in columns preceding the data. To make corrections for this, a sequence designation card is required to rearrange the input sequence. As an example, if the data card format were:

Columns	Description
(1- 3)	Not used
(4- 5)	Data field
(6- 9)	Not used
(10-11)	Treatment number

the format statement would be:

```
(3X, F2.0, 4X, F2.0)
     ↓         ↓
     Data      Treatment number
```

This means that the data is being read in first and the treatment number is being read second. Therefore, the proper sequence designation would be 2,1. This means that the second element being read is the treatment number and the first element is the data.

DATA CARDS

The data cards must be punched so that they can be read with a

Fortran read statement. No alpha-numeric codes may be used for treatment numbers. The treatment numbers must be consecutive integers starting with one. The data fields may or may not be punched with a decimal but the procedure must remain constant for all data cards once the decision is reached. If negative data is required, then an extra column must be allowed for the negative sign. The negative sign is made by using an eleven punch to the left of the most significant digit in the field. The data cards must be sorted blocks within groups. An invalid F-test and erroneous adjusted treatment means will result if this is not done.

IV. OUTPUT:

The cards preceding the data cards are first read and then printed out. The next thing printed out is the completed analysis of variance table. The rest of the output is extensively labeled and no explanation will be given here.

V. COMMENTS:

Even though it is not a requirement for the program to have the block and group numbers punched on each card, it would be advisable to do so for ease in sorting the cards in the previously specified manner.

The program is presently set up to read only one data field at a time. If the user wishes to analyze multiple data fields, then he will first need to put all of the data on either disk or tape by using some of the available utility programs or creating his own program to accomplish this. After the data has been put on disk or tape, the data fields can then be obtained individually by using the appropriate format statement. Batch processing is assumed.

If the option is used to both print and punch the unadjusted, adjusted treatment means, and the standard errors of the difference of two means, the format and sequence for the cards is the same as for what is printed. The first card punched contains the unadjusted and adjusted standard errors of the difference between two means in that order. On the following cards the treatment number is punched first, the unadjusted treatment mean is punched second, and the adjusted treatment mean is punched last. The decimal is also punched in the output cards.

VI. SAMPLE PROBLEM:

The design used is No. 11.15 as found in Cochran and Cox, "Experimental Designs." There were 3 groups, 10 treatments, 30 blocks, 3 units per block, and 3 replications per group for this sample problem. The data cards were punched in the following manner:

Columns	Description
(1- 3)	Not used
(4- 5)	Treatment number
(6)	Not used
(7- 8)	Data (XX.)

The DØ loop reads in the data cards and some basic calculations are performed within the loop as follows:

Total (uncorrected) sums of squares = 25,536

Block (unadjusted) sums of squares = 74,246

Block	Total	Block	Total	Block	Total	Block	Total
1	27	7	24	13	94	19	51
2	24	8	45	14	41	20	65
3	90	9	79	15	34	21	22
4	40	10	21	16	19	22	22
5	46	11	13	17	84	23	17
6	65	12	45	18	26	24	46

Block	Total
25	80
26	18
27	38
28	67
29	20
30	49

Grand total = 1312

Group	Total
1	461
2	472
3	379

The following analysis of variance is then derived:

Source	d.f.	Sums of Squares	Mean Squares
Total	89	64099.560	
Group	2	172.156	
Unadjusted treatment	9	1478.622	
Blocks within groups	27	4344.210	160.900
Error	52	414.968	8.137

It is invalid to make an F-test in the conventional manner using the above information. The adjusted F-ratio that is calculated and printed out last is computed in the following manner. The sum of squares of deviations of the adjusted treatment totals is calculated first. This is then divided by (tr-c) where t is the number of treatments, r is the number of replications, and c is the number of groups. The quotient is then divided by the effective error mean square, which is the error variance/unit in the output. The effective error is estimated as $E_e \left[1 + (t-c)\mu\right]$ where E_e is the error mean square from the above analysis of variance, μ is the weighting factor, and the other terms are the same as previously mentioned.

DESIGNS NOT ARRANGED IN REPLICATIONS OR TREATMENTS (TYPE III)

I. DESCRIPTION:

This program was written to serve as a utility program in a statistical laboratory. The program will compute the analysis of variance, the unadjusted and adjusted treatment means, and other pertinent statistics. At the present time this program can handle up to 100 treatments. If more treatments are needed, then a simple modification of the dimension statement is all that is necessary.

At the user's discretion the program can also punch the unadjusted and adjusted treatment means on cards in addition to having these means printed. It is advisable to have some knowledge of Fortran language.

II. METHODOLOGY:

A DØ loop reads the data cards. The treatment number is read first and the data field second. Some of the calculations performed during this loop are total sums of squares (uncorrected), block sums of squares, block totals, and the grand total of all observations. The other necessary computations are then performed using the procedures described by Cochran and Cox, "Experimental Designs," pp. 446-447.

III. INPUT:

The input consists of three types of cards preceding the data cards. The first card is the control card which defines to the program the number of treatments, the number of times each treatment is replicated, the number of units per block, and the total number of blocks in the experiment. The format card using an appropriate Fortran statement describing how the data cards are to be read comes next. The last card is the sequence designation card.

CONTROL CARD

(6I5, 10X, 10A4)

Columns	Description
(3- 5)	Number of treatments
(9-10)	Number of times each treatment is replicated
(14-15)	Number of units per block
(19-20)	Total number of blocks
(25)	Type of output wanted 0 = Print unadjusted and adjusted treatment means 1 = Print and punch unadjusted, adjusted treatment means, and standard errors of the difference between two treatment means
(30)	Data input device code 1-4 = Tape, disk, etc. (see COMMENTS) 5 = Card reader
(41-80)	Descriptive information

INPUT FORMAT SPECIFICATION

The first column must be a left parenthesis (punched on either an 026 or 029 key punch). This is followed by the Fortran format of the data cards. This consists of any combination of wX and Fw.d specifications. The Fw.d is used to read both the treatment number and the desired data field. Up to 80 columns may be used for the format expression.

SEQUENCE DESIGNATION CARD
(2013)

Columns	Description
(2- 3)	Position of treatment number in read list
(5- 6)	Position of data field in read list

There are times when the treatment number will not be punched in columns preceding the data. To make allowances for this, a sequence designation card is required to rearrange the input sequence. As an example, if the data card format were:

Columns	Description
(1- 5)	Not used
(6- 9)	Data (XX.XX)
(10-13)	Not used
(14-15)	Treatment number

the format statement would be:

```
(5X, F4.2, 4X, F2.0)
     ↓         ↓
     Data      Treatment number
```

This means that the first element being read is the data while the treatment number is being read last. Therefore, the proper sequence designation would be 2,1. This means that the second element read corresponds to the treatment number and the first element is the data.

DATA CARDS

The data cards must be punched so that they are compatible with Fortran read statements. No alpha-numeric codes are used to identify treatment numbers. The treatment numbers must be consecutive integers starting with one. The data fields may or may not be punched with a decimal, but whatever the choice it must be consistent. If negative data is required, then an extra column must be allowed for the negative sign. The negative sign is made by using the eleven punch to the left of the most significant digit in the field. The data cards must be sorted into blocks. If this is not done, the adjusted treatment means and the calculated F-test will not be correct.

IV. OUTPUT:

All cards preceding the data are read and then printed. The next print out is the completed analysis of variance. The rest of the output is extensively labeled and no explanation will be given here.

V. COMMENTS:

Even though it is not a requirement for the program to have the block number punched in the data cards, it would ease the problem somewhat of sorting the cards in blocks.

The program is currently set up to read and analyze only one data field at a time. If the user wished to analyze multiple data fields per card, then he would need to put all of the data on some device, such as a disk or tape, in order to manipulate the program and the data to obtain the desired analysis. There are utility programs available that will store the data on tape or disk. After the data has been put on one of these devices, the individual data fields can then be obtained by using the appropriate format expression. Batch processing is assumed.

If the option to both print and punch the unadjusted, adjusted treatment means, and the standard errors of the difference of two means, the format and sequence for the punched cards is the same as for what is printed. The first card punched contains the unadjusted and adjusted standard errors of the difference between two means in that order. On the following cards the treatment number is punched first, then the unadjusted treatment mean is punched second, and the adjusted treatment mean is punched last. The decimal is punched in the output cards.

VI. SAMPLE PROBLEM:

The data for this example was taken from Fisher and Yates. There were 9 treatments, 4 units per block, 18 total blocks, and each treatment was replicated 8 times. The analysis of variance output by the program is the same as that by Fisher and Yates. The other statistics, however, do not appear in their example. The computation procedures and explanations of these statistics can be found in Cochran and Cox (pp. 443-447).

The data cards were punched in the following format:

Columns	Description
(1- 2)	Treatment number

(3- 5)	Not used
(6- 8)	Data (X.XX)

The adjusted F-ratio is computed by first finding the sum of squares of the adjusted means. Divide this sum of squares by the number of treatments minus one, (t-1). This quotient divided by the effective error variance per unit gives the adjusted F-ratio.

LITERATURE CITED

Cochran, William G. and Gertrude M. Cox. 1950. Experimental Designs. John Wiley & Sons, Inc., New York.

Fisher, R. A. and F. Yates. 1963. Statistical tables for agricultural, biological, and medical research. Hafner, New York.

Hurst, Rex L. n.d., a. Balanced lattice program. Unpublished. Utah State University, Logan, Utah.

Hurst, Rex L. n.d., b. Designs arranged in replications program. Unpublished. Utah State University, Logan, Utah.

Hurst, Rex L. n.d., c. Designs arranged in groups of replications. Unpublished. Utah State University, Logan, Utah.

Hurst, Rex L. n.d., d. Designs not arranged in replications or treatments. Unpublished. Utah State University, Logan, Utah.

APPENDIXES

Appendix A

Listing of Balanced Lattice Program

```
      IMPLICIT REAL * 8(A-H,O-Z)
      DIMENSION IDK(10,81),TT(81),TR(10),BT(81),TB(90),
     1FMT(50),ABC(10)
      IRD=5
      IWRT=6
      IPCH=7
      WRITE(IWRT,102)
  102 FORMAT(1H1,'BALANCED LATTICES'//)
      READ (IRD,100)NT,NR,NPCH,NZ,(ABC(I),I=1,10)
  100 FORMAT (4I5,20X,10A4)
      WRITE(IWRT,101)NT,NR,NPCH,NZ,(ABC(I),I=1,10)
  101 FORMAT(1H ,4I5,20X,10A4)
      READ(IRD,128)(FMT(I),I=1,20)
      WRITE(IWRT,129)(FMT(I),I=1,20)
  128 FORMAT(  20A4)
  129 FORMAT(1H ,20A4)
      READ(IRD,127)N1,N2
  127 FORMAT(2I3)
      WRITE(IWRT,124)N1,N2
  124 FORMAT(1H ,2I3)
      SS=0.0
      G=0.0
      NK=NR-1
      NB=NR*NT/NK
      IF(NZ.NE.IRD) REWIND NZ
      DO 4 J=1,NT
      TT(J)=0.0
    4 BT(J)=0.0
      DO 5 K=1,NB
    5 TB (K)=0.0
      DO 7  I=1,NR
      TR (I)=0.0
      DO 7  J=1,NT
      READ(NZ,FMT)AB,AD
      IF(1-N1)3,6,97
    3 Y=AB
      K=AD
      GO TO 2
    6 Y=AD
      K=AB
    2 IDK(I,J)=K
      SS=SS+Y*Y
      K=IDK(I,J)
      G=G+Y
      TT(J)=TT(J)+Y
      TR(I)=TR(I)+Y
    7 TB(K)=TB(K)+Y
      DO 8  I=1,NR
      DO 8 J=1,NT
```

```
      K=IDK(I,J)
    8 BT(J)=BT(J)+TB(K)
      BLK=0.0
      DO 9 J=1,NT
      BT(J)=DFLOAT(NK)*TT(J)-DFLOAT(NR)*BT(J)+G
    9 BLK=BLK+BT(J)*BT(J)
      BLK=BLK/DFLOAT((NK**3)*(NK+1))
      TOT=SS-G*G/DFLOAT(NR*NT)
      REP=0.0
      DO 10 I=1,NR
   10 REP=REP+TR(I)*TR(I)/DFLOAT(NT)
      REP=REP-G*G/DFLOAT(NR*NT)
      TRT=0.0
      DO 11 J=1,NT
   11 TRT=TRT+TT(J)*TT(J)/DFLOAT(NR)
      TRT=TRT-G*G/DFLOAT(NR*NT)
      ERR=TOT-REP-TRT-BLK
      NTOT=NR*NT-1
      NREP=NR-1
      NTRT=NT-1
      NBLK=NT-1
      NERR=NTOT-NREP-NTRT-NBLK
      AMTRT=TRT/DFLOAT(NTRT)
      AMBLK=BLK/DFLOAT(NBLK)
      AMERR=ERR/DFLOAT(NERR)
      AMU=(AMBLK-AMERR)/(DFLOAT(NK*NK)*AMBLK)
      WRITE(IWRT,103)
  103 FORMAT (1H0,'ANALYSIS OF VARIANCE'/)
      WRITE(IWRT,104)
  104 FORMAT(1H ,'SOURCE',6X,'DF',2X,'SUMS OF SQUARES',2X,
     1'MEAN SQUARES')
      WRITE (IWRT,105)NTOT,TOT
  105 FORMAT(1H ,'TOT     ',I6,E17.8)
      WRITE (IWRT,117)NREP,REP
  117 FORMAT (1H ,'REP     ',I6,E17.8)
      WRITE (IWRT,106)NTRT,TRT,AMTRT
  106 FORMAT (1H ,'TRT     ',I6,2E17.8)
      WRITE (IWRT,107)NBLK,BLK,AMBLK
  107 FORMAT(1H ,'BLK(A)  ',I6,2E17.8)
      WRITE(IWRT,108)NERR,ERR, AMERR
  108 FORMAT(1H ,'ERROR   ',I6,2E17.8//)
      WRITE(IWRT,110)
  110 FORMAT(1H ,'TRT',5X,'UNADJ MEANS',6X,'ADJ   MEANS'/)
      DO 12 J=1,NT
      BT(J)=(TT(J)+AMU*BT(J))/DFLOAT(NR)
      TT(J)=TT(J)/DFLOAT(NR)
   12 WRITE(IWRT,111)J,TT(J),BT(J)
  111 FORMAT(I5,2E17.8)
      AVE=G/DFLOAT(NR*NT)
```

```
      WRITE(IWRT,112)AVE
  112 FORMAT(1H ,'EXPERIMENT AVE',E15.8)
      GO TO 98
   97 WRITE(IWRT,126)
  126 FORMAT(1H0,' THERE IS AN ERROR IN THE CONTROL CARD')
      GO TO 91
   98 EF=AMERR*(1.+DFLOAT(NK)*AMU)
      WRITE(IWRT,115)EF
  115 FORMAT(1H ,'EFFECTIVE ERROR=',E15.8)
      CV=DSQRT(EF)/AVE
      WRITE(IWRT,113)CV
  113 FORMAT(1H ,'COEFFICIENT OF VARIATION=',E15.8)
      IF(1-NPCH)97,99,999
   99 SEUM=DSQRT(2.*AMERR/DFLOAT(NR))
      SEF=DSQRT(2*EF/DFLOAT(NR))
      WRITE(IPCH,131)SEUM,SEF
  131 FORMAT(6X,2F17.8)
      DO 61 I=1,NT
   61 WRITE(IPCH,125)I,BT(I),TT(I)
  125 FORMAT(I4,2X,2F17.8)
  999 ATS=0.0
      DO 15 J=1,NT
   15 ATS=ATS+((BT(J)-AVE)*(BT(J)-AVE))
      F=ATS*DFLOAT(NR)/(DFLOAT(NTRT)*EF)
      WRITE(IWRT,116)F
  116 FORMAT(1H ,'ADJ F RATIO=',E15.8//)
   91 CALL EXIT
      END
```

Appendix B

Input and Output for Balanced Lattice Sample Problem

SAMPLE INPUT

```
    9    4         1                         BALANCED LATTICE TEST
(F2.0,3X,F3.2)
  1  2
 111 220
 112 184
 113 218
 214 205
 215  85
 216 186
 317  73
 318 160
 319 176
 421 119
 522 226
 623 212
 424 120
 525 107
 626 203
 427 115
 528 145
 629 163
 731 181
 832 176
 933 171
 934 157
 735 116
 836 216
 837 180
 938 113
 739 111
1041 177
1142 150
1243 204
1144 160
1245  93
1046 157
1247 178
1048 143
1149 142
```

BALANCED LATTICES

```
    9    4    0    5                        BALANCED LATTICE TEST
(F2.0,3X,F3.2)
  1  2
```

ANALYSIS OF VARIANCE

SOURCE	DF	SUMS OF SQUARES	MEAN SQUARES
TOT	35	0.59609000D 01	
REP	3	0.77388889D-01	
TRT	8	0.32261000D 01	0.40326250D 00
BLK(A)	8	0.14206037D 01	0.17757546D 00
ERROR	16	0.12368074D 01	0.77300463D-01

TRT	UNADJ MEANS	ADJ MEANS
1	0.17425000D 01	0.18035178D 01
2	0.18400000D 01	0.17543554D 01
3	0.20125000D 01	0.19643445D 01
4	0.16050000D 01	0.17267220D 01
5	0.10025000D 01	0.93928615D 00
6	0.19050000D 01	0.18447665D 01
7	0.13650000D 01	0.13869601D 01
8	0.14025000D 01	0.14346559D 01
9	0.14800000D 01	0.15003916D 01

EXPERIMENT AVE 0.15950000D 01
EFFECTIVE ERROR= 0.91850716D-01
COEFFICIENT OF VARIATION= 0.19001181D 00
ADJ F RATIO= 0.43163811D 01

Appendix C

Listing of Designs Arranged in Replications Program

```
      IMPLICIT REAL * 8(A-H,O-Z)
      DIMENSION TT(100),BT(100),NID(100),FMT(40),X(100),
     1Y(100),ABC(10)
      IRD=5
      IWRT=6
      IPCH=7
      WRITE(IWRT,102)
  102 FORMAT(1H1,'INC. BLK.-ARRANGED IN REPLICATIONS'//)
    1 READ(IRD,100)NT,NK,NR,NB,NPCH,NZ,(ABC(I),I=1,10)
  100 FORMAT(6I5,10X,10A4)
      WRITE(IWRT,101)NT,NK,NR,NB,NPCH,NZ,(ABC(I),I=1,10)
  101 FORMAT(1H ,6I5,10X,10A4)
      READ(IRD,128)(FMT(I),I=1,20)
      WRITE(IWRT,129)(FMT(I),I=1,20)
  128 FORMAT(   20A4)
  129 FORMAT(1H ,20A4)
      READ(IRD,127)N1,N2
  127 FORMAT(2I3)
      WRITE(IWRT,124)N1,N2
  124 FORMAT(1H ,2I3)
      IF(NZ.NE.IRD) REWIND NZ
      DO 4 I=1,100
      NID(I)=0
      BT(I)=0.0
    4 TT(I)=0.0
      SS=0.0
      CT=0.0
      REP=0.0
      BTOT=0.0
      QSQ=0.0
      DO 10 III=1,NR
      REPI=0.0
      DO 20 I=1,NB
      B=0.0
      DO 30 JJ=1,NK
      READ(NZ,FMT)AB,AD
      IF(1-N1)3,5,97
    3 Z=AB
      K=AD
      GO TO 2
    5 Z=AD
      K=AB
      GO TO 2
    2 NID(JJ)=K
      B=B+Z
      TT(K )=TT(K )+Z
      REPI=REPI+Z
      CT=CT+Z
   30 SS=SS+Z*Z
```

```
      BTOT=BTOT + B*B
      DO20 II=1,NK
      I1= NID(II)
   20 BT(I1)=BT(I1)+B
   10 REP=REP+REPI*REPI
      G=DFLOAT(NK-1)*CT
      CT=CT*CT/DFLOAT(NR*NT)
      SS=SS-CT
      SST=SS
      WRITE(IWRT,130)
  130 FORMAT(1HO,'ANALYSIS OF VARIANCE'/)
      WRITE(IWRT,103)
  103 FORMAT(1H ,'SOURCE',6X,'DF',2X,'SUMS OF SQUARES',2X,
     1'MEAN SQUARES')
      NDF=NT*NR-1
      WRITE(IWRT,104)NDF,SS
      SS=0.0
  104 FORMAT(1H ,'TOT       ',I6,E17.8)
      DO 40 I=1,NT
   40 SS= SS +TT(I)*TT(I)
      SS=SS/DFLOAT(NR)-CT
      REP=REP/NT-CT
      NDF=NR-1
      WRITE(IWRT,105)NDF,REP
  105 FORMAT(1H ,'REP       ',I6,E17.8)
      NDF=NT-1
      WRITE(IWRT,106)NDF,SS
  106 FORMAT(1H ,'UNAJ TRT ',I5,2E17.8)
      DO 50I=1,NT
      Q=DFLOAT(NK)*TT(I)-BT(I)
      QSQ=QSQ+Q*Q
      W=DFLOAT(NT-NK)*TT(I)-DFLOAT(NT-1)*BT(I)+G
   50 BT(I)=W
      SSBWR=(DFLOAT(NT-1)*QSQ)/ DFLOAT(NK*NT*NR*(NK-1))+
     2BTOT/DFLOAT(NK)-CT-SS-REP
      NDF=NR*(NB-1)
      SMBWR=SSBWR/DFLOAT(NR*(NB-1))
      WRITE(IWRT,107)NDF,SSBWR,SMBWR
  107 FORMAT(1H ,'BLK/REP ',I6,2E17.8)
      SSER=SST-SS-REP-SSBWR
      SMER=SSER/DFLOAT(NT*NR-NT-NB*NR+1)
      NDF=NT*NR-NT-NB*NR+1
      WRITE(IWRT,108)NDF,SSER,SMER
  108 FORMAT(1H ,'ERROR     ',I6,2E17.8/)
      WF=DFLOAT(NR)*(SMBWR-SMER)
      XI=NR*NT*(NK-1)
      XJ=NK*(NB*NR-NR-NT+1)
      WF=WF/(XI*SMBWR+XJ*SMER)
      WRITE(IWRT,110)
```

```
  110 FORMAT(1H ,'TREATMENT MEANS')
      WRITE(IWRT,111)
  111 FORMAT(1H ,'  TRT',3X,'UNADJ MEANS',6X,'ADJ MEANS')
      DO 60 I=1,NT
      X(I)=TT(I)/DFLOAT(NR)
      Y(I)=(TT(I)+WF*BT(I))/DFLOAT(NR)
   60 WRITE (IWRT,112)I,X(I),Y(I)
  112 FORMAT(1H ,I5,2E17.8)
      GO TO 98
   97 WRITE(IWRT,126)
  126 FORMAT(1HO,' THERE IS AN ERROR IN THE CONTROL CARD')
      GO TO 9
   98 XX=1+(NT-NK)*WF
      ERVAR=SMER*XX
      WRITE(IWRT,113)ERVAR
  113 FORMAT(1H , 'ERROR VAR./UNIT     ='E15.8)
      XA=DSQRT(2.*ERVAR/DFLOAT(NR))
      WRITE(IWRT,114)XA
  114 FORMAT(1H ,'ST. ER. ADJ. MEANS ='E15.8)
      IF(1-NPCH)97,99,999
   99 SEUM=DSQRT(2.*SMER/DFLOAT(NR))
      WRITE(IPCH,131)SEUM,XA
  131 FORMAT(6X,2F17.8)
      DO 61 I=1,NT
   61 WRITE(IPCH,125)I,X(I),Y(I)
  125 FORMAT(I4,2X,2F17.8)
  999 F=0.0
      DO 70 I=1,NT
   70 F=F+(TT(I)+WF*BT(I))*(TT(I)+WF*BT(I))
      F=((F/DFLOAT(NR)-CT)/DFLOAT(NT-1))/ERVAR
      WRITE(IWRT,115)F
  115 FORMAT(1H ,'ADJUSTED F RATIO   ='E15.8)
    9 CALL EXIT
      END
```

Appendix D

Input and Output for Designs Arranged in Replications Program

SAMPLE INPUT

```
    6    2    5    3         5          MILLER TEST DATA
(1X,F1.0,1X,F2.0)
  1  2
 1  7
 2 17
 3 26
 4 25
 5 33
 6 29
 1 17
 3 27
 2 23
 5 27
 4 29
 6 30
 1 10
 4 25
 2 26
 6 37
 3 24
 5 26
 1 25
 5 40
 2 25
 4 34
 3 34
 6 32
 1 11
 6 27
 2 24
 3 21
 4 26
 5 32
```

INC. BLK.-ARRANGED IN REPLICATIONS

```
    6    2    5    3    0    5          MILLER TEST DATA
(1X,F1.0,1X,F2.0)
  1  2
```

ANALYSIS OF VARIANCE

SOURCE	DF	SUMS OF SQUARES	MEAN SQUARES
TOT	29	0.164896670 04	
REP	4	0.298466670 03	
UNAJ TRT	5	0.105976670 04	
BLK/REP	10	0.213400000D 03	0.213400000D 02
ERROR	10	0.773333330 02	0.773333330 01

TREATMENT MEANS

TRT	UNADJ MEANS	ADJ MEANS
1	0.140000000 02	0.143602990 02
2	0.230000000 02	0.234551150 02
3	0.264000000 02	0.267223730 02
4	0.278000000 02	0.280844470 02
5	0.316000000 02	0.311448850 02
6	0.310000000 02	0.300328810 02

ERROR VAR./UNIT = 0.106662950 02
ST. ER. ADJ. MEANS = 0.206555510 01
ADJUSTED F RATIO = 0.176962220 02

Appendix E

Listing of Designs Arranged in Groups of Replications

```
      IMPLICIT REAL * 8(A-H,O-Z)
      DIMENSION TT(100),BT(100),NID(100),FMT(50),X(100),
     3Y(100),ABC(10)
      IRD=5
      IWRT=6
      IPCH=7
      WRITE(IWRT,102)
  102 FORMAT(1H1,'A.O.V.   INC. BLK.-ARRANGED IN GROUPS'//)
      READ(IRD ,100)NT,NK,NR,NB,NC,NPCH,NZ,(ABC(I),I=1,10)
      WRITE(IWRT,101)NT,NK,NR,NB,NC,NPCH,NZ,(ABC(I),I=1,10)
  100 FORMAT(7I5,5X,10A4)
  101 FORMAT(1H ,7I5,5X,10A4)
      READ(IRD,128)(FMT(I),I=1,20)
      WRITE(IWRT,129)(FMT(I),I=1,20)
  128 FORMAT(   20A4)
  129 FORMAT(1H ,20A4 )
      READ(IRD,127)NA,N2
  127 FORMAT(2I3)
      WRITE(IWRT,124)NA,N2
  124 FORMAT(1H ,2I3)
      IF(NZ.NE.IRD) REWIND NZ
      DO 4 I=1,100
      NID(I)=0
      BT(I)=0.0
    4 TT(I)=0.0
      SS=0.0
      CT=0.0
      GRP=0.0
      QSQ=0.0
      BTOT=0.0
      F=0.0
      DO 30 III=1,NC
      GRPI=0.0
      DO 20 II=1,NB
      B=0.0
      DO 10 I=1,NK
      READ(NZ,FMT)AB,AD
      IF(1-NA)3,5,97
    3 Z=AB
      K=AD
      GO TO 2
    5 Z=AD
      K=AB
      GO TO 2
    2 NID(I)=K
      N1=NID(I)
      B=B+Z
      TT(K)=TT(K)+Z
      GRPI=GRPI+Z
```

```
      CT=CT+Z
   10 SS=SS+Z*Z
      BTOT=BTOT+B*B
      DO 20 N2=1,NK
      N3=NID(N2)
   20 BT(N3)=BT(N3)+B
   30 GRP=GRP+GRPI*GRPI
      G=DFLOAT(NK-1)*CT
      CT=CT*CT/DFLOAT(NR*NT)
      SS=SS-CT
      SST=SS
      WRITE(IWRT,130)
  130 FORMAT(1H0,'ANALYSIS OF VARIANCE'/)
      WRITE(IWRT,103)
  103 FORMAT(1H0,'SOURCE',6X,'DF',2X,'SUMS OF SQUARES',2X,
     4'MEAN SQUARES')
      NDF=NT*NR-1
      WRITE(IWRT,104)NDF,SS
  104 FORMAT(1H ,'TOT       ',I6,E17.8)
      SS=0.0
      DO 40 I=1,NT
   40 SS=SS+TT(I)*TT(I)
      SS=SS/DFLOAT(NR)-CT
      GRP=GRP/DFLOAT(NB*NK)-CT
      NDF=NC-1
      WRITE(IWRT,105)NDF,GRP
  105 FORMAT(1H ,'GRP       ',I6,E17.8)
      NDF=NT-1
      WRITE(IWRT,106)NDF,SS
  106 FORMAT(1H ,'UNAJ TRT ',I5,E17.8)
      DO 50 I=1,NT
      Q=DFLOAT(NK)*TT(I)-BT(I)
      QSQ=QSQ+Q*Q
      W=DFLOAT(NT-NK)*TT(I)-DFLOAT(NT-1)*BT(I)+G
   50 BT(I)=W
      XX=(NT-1)*QSQ
      SSBWG=XX /DFLOAT(NK*NT*NR*(NK-1))+BTOT/DFLOAT(NK)-CT-
     5SS-GRP
      NDF=NC*(NB-1)
      SMBWG=SSBWG/DFLOAT(NDF)
      WRITE(IWRT,107)NDF,SSBWG,SMBWG
  107 FORMAT(1H ,'BLK/GRP ',I6,2E17.8)
      SSER=SST-SS-SSBWG-GRP
      NDF=NT*NR-NT-NB*NC+1
      SMER=SSER/DFLOAT(NT*NR-NT-NB*NC+1)
      WRITE(IWRT,108)NDF,SSER,SMER
  108 FORMAT(1H ,'ERROR    ',I6,2E17.8/)
      WF=DFLOAT(NB*NC-NC)*(SMBWG-SMER)
      XY=NT*(NK-1)*(NB*NC-NC)*SMBWG+(NT-NK)*(NB*NC-NC-NT+1)*
```

```
     6SMER
      WF=WF/XY
      WRITE(IWRT,110)
  110 FORMAT(1H ,'TREATMENT MEANS')
      WRITE(IWRT,111)
  111 FORMAT(1H ,2X,'TRT',3X,'UNADJ MEANS',6X,'ADJ MEANS')
      DO 60 I=1,NT
      X(I)=TT(I)/DFLOAT(NR)
      Y(I)=(TT(I)+WF*BT(I))/DFLOAT(NR)
   60 WRITE (IWRT,112)I,X(I),Y(I)
  112 FORMAT(1H ,I5,2E17.8)
      GO TO 98
   97 WRITE(IWRT,126)
  126 FORMAT(1HO,' THERE IS AN ERROR IN THE CONTROL CARD')
      GO TO 9
   98 XZ=1+(NT-NK)*WF
      ERVAR=SMER*XZ
      WRITE(IWRT,113) ERVAR
  113 FORMAT(1H , 'ERROR VAR./UNIT     = ',E15.8)
      XA=DSQRT(2.*ERVAR/DFLOAT(NR))
      WRITE(IWRT,115)XA
  115 FORMAT(1H ,'ST. ER. ADJ. MEANS = ',E15.8)
      IF(1-NPCH)97,99,999
   99 SEUM=DSQRT(2.*SMER/DFLOAT(NR))
      WRITE(IPCH,131)SEUM,XA
  131 FORMAT(6X,2F17.8)
      DO 61 I=1,NT
   61 WRITE(IPCH,125)I,X(I),Y(I)
  125 FORMAT(I4,2X,2F17.8)
  999 DO 70 I=1,NT
   70 F=F+(TT(I)+WF*BT(I))*(TT(I)+WF*BT(I))
      YY=(F/NR-CT)/(NT-1)
      F=YY/ERVAR
      WRITE(IWRT,116)F
  116 FORMAT(1H ,'ADJUSTED F RATIO    = ',E15.8)
    9 CALL EXIT
      END
```

Appendix F

Input and Output for Designs Arranged in Groups of Replications

SAMPLE INPUT

```
  10    3    9   10    3          5     MILLER TEST DATA
(3X,F2.0,1X,F2.0)
  1  2
101 3 11
101 2  9
101 1  7
102 8 12
102 5  8
102 2  4
103 3 27
103 4 29
103 7 34
104 6 15
104 1 11
104 4 14
105 5 13
105 7 16
105 8 17
106 9 25
106 4 18
106 6 22
107 1  3
107 9 12
107 7  9
10810 19
108 8 16
108 2 10
109 3 20
109 9 28
10910 31
11010 10
110 5  4
110 6  7
211 1  2
211 2  5
211 4  6
212 2 13
212 6 17
212 3 15
213 3 29
213 4 30
213 8 35
214 5  9
214 9 19
214 4 13
215 7 15
215 5 12
215 1  7
```

216	9	8
216	6	4
216	8	7
217	7	29
217	3	23
217	10	32
218	10	13
218	1	3
218	8	10
219	2	12
219	9	21
219	5	18
220	6	22
220	7	19
220	10	24
321	3	7
321	5	10
321	1	5
322	2	5
322	6	7
322	7	10
323	8	6
323	9	7
323	3	4
324	10	18
324	4	16
324	2	12
325	5	27
325	3	24
325	6	29
326	8	3
326	1	10
326	6	5
327	2	10
327	9	15
327	7	13
328	8	20
328	4	18
328	7	29
329	1	9
329	9	3
329	10	8
330	4	20
330	10	11
330	5	18

A.O.V. INC. BLK.-ARRANGED IN GROUPS

```
   10    3    9   10    3    0    5      MILLER TEST DATA
(3X,F2.0,1X,F2.0)
  1  2
```

ANALYSIS OF VARIANCE

SOURCE	DF	SUMS OF SQUARES	MEAN SQUARES
TOT	89	0.64099556D 04	
GRP	2	0.17215556D 03	
UNAJ TRT	9	0.14786222D 04	
BLK/GRP	27	0.43442111D 04	0.16089671D 03
ERROR	51	0.41496667D 03	0.81366013D 01

TREATMENT MEANS

TRT	UNADJ MEANS	ADJ MEANS
1	0.63333333D 01	0.11582921D 02
2	0.88888889D 01	0.10895931D 02
3	0.17777778D 02	0.12053798D 02
4	0.18222222D 02	0.14192499D 02
5	0.13222222D 02	0.13978121D 02
6	0.14222222D 02	0.14978121D 02
7	0.19333333D 02	0.17310652D 02
8	0.14000000D 02	0.15574355D 02
9	0.15333333D 02	0.17439424D 02
10	0.18444444D 02	0.17771955D 02

ERROR VAR./UNIT = 0.10808865D 02
ST. ER. ADJ. MEANS = 0.15498290D 01
ADJUSTED F RATIO = 0.52195577D 01

Appendix G

Listing of Designs Not Arranged in Replications or Treatments

```
      IMPLICIT REAL * 8(A-H,O-Z)
      DIMENSION TT(100),BT(100),N1D(100),FMT(50),X(100),
     7Y(100),ABC(10)
      IRD=5
      IWRT=6
      IPCH=7
      WRITE(IWRT,102)
  102 FORMAT(1H1,'AOV  INC BLK-NOT ARRANGED IN BLKS OR REPS'//)
      READ(IRD,100)NT,NR,NK,NB,NPCH,NZ,(ABC(I),I=1,10)
  100 FORMAT(6I5,10X,10A4)
      WRITE(IWRT,101)NT,NR,NK,NB,NPCH,NZ, (ABC(I),I=1,10)
  101 FORMAT(1H ,6I5,10X,10A4)
      READ(IRD,128)(FMT(I),I=1,20)
      WRITE(IWRT,129)(FMT(I),I=1,20)
  128 FORMAT(  20A4)
  129 FORMAT(1H ,20A4)
      READ(IRD,127)N1,N2
  127 FORMAT(2I3)
      WRITE(IWRT,124)N1,N2
  124 FORMAT(1H ,2I3)
      IF(NZ.NE.IRD) REWIND NZ
      DO 4I=1,100
      N1D(I)=0
      BT(I)=0.0
    4 TT(I)=0.0
      SS=0.0
      CT=0.0
      QSQ=0.0
      BTOT=0.0
      F=0.0
      SST=0.0
      DO 20 I1=1,NB
      B=0.0
      DO 10 I2=1,NK
      READ(NZ,FMT)AB,AD
      IF(1-N1)3,5,97
    3 Z=AB
      K=AD
      GO TO 2
    5 Z=AD
      K=AB
      GO TO 2
    2 B=B+Z
      N1D(I2)=K
      TT(K)=TT(K)+Z
      CT=CT+Z
   10 SS=SS+Z*Z
      BTOT=BTOT+B*B
      DO 20 I4=1,NK
```

```
      I5=NID(I4)
   20 BT(I5)=BT(I5)+B
      G=DFLOAT(NK-1)*CT
      CT=CT*CT/DFLOAT(NK*NB)
      SS=SS-CT
      WRITE(IWRT,130)
  130 FORMAT(1H0,'ANALYSIS OF VARIANCE'/)
      WRITE(IWRT,103)
  103 FORMAT(1H0,'SOURCE',6X,'DF',2X,'SUMS OF SQUARES',3X,
     8'MEAN SQUARES')
      NDF=NK*NB-1
      WRITE(IWRT,104)NDF,SS
  104 FORMAT(1H ,'TOT      ',I6,E17.8)
      DO 30 I=1,NT
   30 SST=SST+TT(I)*TT(I)
      SST=SST/DFLOAT(NR)-CT
      NDF=NT-1
      WRITE(IWRT,106)NDF,SST
  106 FORMAT(1H ,'UNAJ TRT ',I5,E17.8)
      DO 40 I=1,NT
      Q=DFLOAT(NK)*TT(I)-BT(I)
      QSQ=QSQ+Q*Q
      W=(NT-NK)*TT(I)-(NT-1)*BT(I)+G
   40 BT(I)=W
      SSB=((NT-1)*QSQ)/(NK*NT*NR*(NK-1))+BTOT/NK-CT-SST
      NDF=NB-1
      SSBER=SSB/DFLOAT(NB-1)
      WRITE(IWRT,107)NDF,SSB,SSBER
  107 FORMAT(1H ,'ADJ BLK ',I6,2E17.8)
      SSER=SS-SST-SSB
      NDF=NT*NR-NT-NB+1
      SMER=SSER/DFLOAT(NT*NR-NT-NB+1)
      WRITE(IWRT,108)NDF,SSER,SMER
  108 FORMAT(1H ,'ERROR    ',I6,2E17.8/)
      WF=DFLOAT(NB-1)*(SSBER-SMER)
      XX=NT*(NK-1)*(NB-1)*SSBER+(NT-NK)*(NB-NT)*SMER
      WF=WF/XX
      WRITE(IWRT,110)
  110 FORMAT(1H ,'TREATMENT MEANS')
      WRITE(IWRT,111)
  111 FORMAT(1H ,'  TRT',3X,'UNADJ MEANS',6X,'ADJ MEANS')
      DO 60 I=1,NT
      X(I)=TT(I)/DFLOAT(NR)
      Y(I)=(TT(I)+WF*BT(I))/DFLOAT(NR)
   60 WRITE (IWRT,112)I,X(I),Y(I)
  112 FORMAT(1H ,I5,2F17.8)
      GO TO 98
   97 WRITE(IWRT,126)
  126 FORMAT(1H0,' THERE IS AN ERROR IN THE CONTROL CARD')
```

```
      GO TO 9
   98 ERVAR=SMER*(1.+DFLOAT(NT-NK)*WF)
      WRITE(IWRT,113)ERVAR
  113 FORMAT(1H , 'ERROR VAR./UNIT     = ',E15.8)
      XA=DSQRT(2.*ERVAR/DFLOAT(NR))
      WRITE(IWRT,115)XA
  115 FORMAT(1H ,'ST. ER. ADJ. MEANS = ',E15.8)
      IF(1-NPCH)97,99,999
   99 SEUM=DSQRT(2.*SMER/DFLOAT(NR))
      WRITE(IPCH,131)SEUM,XA
  131 FORMAT(6X,2F17.8)
      DO 61 I=1,NT
   61 WRITE(IPCH,125)I,X(I),Y(I)
  125 FORMAT(I4,2X,2F17.8)
  999 DO 70 I=1,NT
   70 F=F+(TT(I)+WF*BT(I))*(TT(I)+WF*BT(I))
      F=        (F/NR-CT)/(NT-1) /ERVAR
      WRITE(IWRT,116)F
  116 FORMAT(1H ,'ADJUSTED F RATIO   = ',E15.8)
    9 CALL EXIT
      END
```

Appendix H

Input and Output for Designs

Not Arranged in Replications or Treatments

SAMPLE INPUT

```
     9     8     4     18            5              MILLER TEST DATA
(F1.0,1X,F3.1)
   1  2
6   26
4   97
3   54
5   69
6   59
7   26
9   59
2   63
1   70
6   46
9   49
3   33
9   24
4   40
7   30
6   24
9   50
8   74
5  103
3   94
4  101
1   97
6   57
8   75
2   39
4   41
5   64
9   63
8   40
6   61
7   44
3   33
2   28
6   26
5   28
8   33
2   57
8   93
3   54
9   61
2   47
7   66
1   55
8   53
1   30
8   14
```

9	42
4	28
3	75
7	22
5	26
1	44
3	37
1	52
4	24
2	24
9	30
7	26
1	47
5	24
4	45
2	60
7	46
3	33
7	26
5	49
4	60
8	46
2	73
5	54
6	57
1	44

AOV INC BLK-NOT ARRANGED IN BLKS OR REPS

```
    9    8    4   18    0    5          MILLER TEST DATA
(F1.0,1X,F3.1)
  1  2
```

ANALYSIS OF VARIANCE

SOURCE	DF	SUMS OF SQUARES	MEAN SQUARES
TOT	71	0.31872444D 03	
UNAJ TRT	8	0.24009444D 02	
ADJ BLK	17	0.17526444D 03	0.10309673D 02
ERROR	46	0.11945056D 03	0.25967512D 01

TREATMENT MEANS

TRT	UNADJ MEANS	ADJ MEANS
1	0.54875000D 01	0.57075428D 01
2	0.48875000D 01	0.49128506D 01
3	0.51625000D 01	0.48680955D 01
4	0.54500000D 01	0.56784930D 01
5	0.52125000D 01	0.52060779D 01
6	0.44500000D 01	0.42647719D 01
7	0.35750000D 01	0.39123314D 01
8	0.53500000D 01	0.50025284D 01
9	0.47250000D 01	0.47473085D 01

ERROR VAR./UNIT   = 0.29478397D 01
ST. ER. ADJ. MEANS = 0.85846370D 00
ADJUSTED F RATIO   = 0.93676389D 00