The BEAST: A Classroom Exercise in Applied Micro-economics

Richard L. Schmalensee

August 1968 343-68

TABLE OF CONTENTS

I

The Nature of the BEAST

The BEAST (Business Economics Analytical and Statistical Task) was born in the fall of 1967. Professor Paul MacAvoy and the author were teaching a managerial economics course that is required of all undergraduates majoring in management. We wanted to devise an exercise that would require our students to apply the techniques of linear regression and marginal analysis to a moderately complex problem. A baby BEAST was created to fill this need. The BEAST attained its current maturity in the summer of 1968. Its growth was greatly encouraged by financial support from the Sloan School of Management and by discussions with Paul MacAvoy.

As it is currently envisioned, BEAST may consist of up to five distinct but connected sub-tasks:

1. The estimation of an industry demand curve of the form Q = Q(P, Y), where Q is sales in units, P is the price per unit, and Y is average consumer income.

2. The estimation of a firm production function of the form Q = Q(K, L), where K and L are the machines and man-hours used by the firm in a particular period.

3. The calculation of competitive price and output, given particular values for the number of firms (n), the wage rate ([illegible]), K, and Y.

4. The calculation of monopoly price and output given values for these same quantities. (It may be specified that the N firms have merged into one.)

5. To determine whether (a) the competitive industry in sub-task 3 or (b) the monopoly in sub-task 4 will expand or contract capacity, given the rental rate for machines (r).

Sub-tasks 1 and 2 are exercises in the use of linear regression techniques. They are based on the program GBEAST, discussed in detail in the next chapter. Briefly, GBEAST generates observations on the variables Y, X_1, and X_2, so that $Y = f(X_1, X_2, u)$, where f is a functional form not revealed to the student, and u is a normally distributed error term. GBEAST also provides information on the likely results of a regression run on the generated data.

Sub-tasks 3, 4, and 5 are solved by SBEAST. As a design aid, it also indicates how equilibrium P and Q will be affected by changes in N, w, K, and Y. Chapter III will discuss SBEAST.

Chapters II and III discuss the mechanics of using these programs in great detail. The reader may wish to skim these chapters on a first reading.

Chapter IV of this paper deals with broader issues of design and administration of the BEAST. The TSP regression package is ideally suited to this exercise and is discussed in both chapters II and IV. The reader will find a copy of the TSP writeup (which may be obtained from the Sloan School Computer Facility) quite helpful. It is also suggested that all students be given a copy of that writeup.

Appendix A discusses GRNUM, the subroutine used by GBEAST to obtain normally distributed random numbers. Appendicees B and C provide listings of GBEAST and SBEAST, as written in Fortran II.

All computation to date has been performed on the Sloan School's IBM 1620 II computer, and all descriptions herein pertain to BEAST as implemented on that system. Since both GBEAST and SBEAST have been written in Fortran II, however, conversion to most other machines should pose not special problems. In addition, TSP has been programmed for some other computers. Information on this can be obtained from R. E. Hall, Department of Economics, University of California, Berkley.

Beast has not been designed to be a rigidly specified problem. This manual attempts to provide tools and suggestions, not commandments. BEAST is a set of problems, whose members are determined by the purposes and ingenuity of the user. The BEAST is to be trained to do your bidding, not the other way around.

II

The GBEAST Program

GBEAST is really the heart of this package. It generates the data from which all of the parameters are estimated by the student. It is designed to aid the user in generating "reasonable" data.

One functional form and the same program are used to generate data for both sub-tasks involving regression analysis. For sub-task 1, the dependent variable is quantity demanded, and the independent variables are price and income. For the production function (sub-task 2), the dependent variable is quantity produced, and the independent variables are capital and labor services used. The demand function chosen here has constant price and income elasticities, and the production function is Cobb-Douglas. Rationalizations of both are given to the students in class, without revealing the exact nature of the underlying forms.

These forms are advantageous for two non-economic reasons. Most obvious is the need for only one fairly simple generation program. Second, as we will see explicitly in chapter III, use of log-linear relations for both demand and production means that sub-tasks 3-5 of BEAST require only the solution of linear equations.

The simplest way to describe the workings of GBEAST is to first outline its input requirements and then to discuss the output it produces. The following assumes that GBEAST has been loaded onto the disk as a main program, and that GRNUM and OWNOUT have been loaded

as subrountines. GRNUM generates normally distributed random numbers; it is described and listed in Appendix A. GBEAST is listed in Appendix B.

Input

The first three cards in the deck must a job card, a card with "+ +XEQSGBEAST" punched in columns 1-12, and a comment card with any desired message punched in columns 2-80. This comment will be printed at the top of every page of output. The last card in the deck must have columns 1-35 blank.

In between, there may be one or more coefficient decks. The first card in each such deck is a coefficient card, punched as follows:[1]

GBEAST Coefficient Card

Columns	Variable	Explanation
1-2	NT	The number of trials to be made with these coefficients. Right-justified, no decimal point.
3	IOC	See test
4	ITC	See text.
5-14	a	
15-24	b1	The numerical values of the coefficients to be used to generate data: a is the scale factor, and b_1 and b_2 are the elasticities of Y with respect to X_1 and X_2.
25-34	b2	

---

[1] On all input cards to GBEAST and SBEAST, a decimal point must be punched in all numerical fields with two or more columns, unless a specific statement to the contrary is made.

The variables IOC and ITC require some explanation. If IOC is zero, any data decks created using these coefficients will be punched, one observation to a card, according to the format (1X,I3,3F15.4), where these fields refer to the number of the observation and the values of Y, X_1, and X_2 corresponding to that observation. If IOC is any number other than zero, the subroutine OWNOUT will be called to punch the cards.

This feature is included because it may be desirable to give the students such variables as revenue (price times quantity) or output per manhour(production divided by labor services), to make sure they know what variables should be present in demand and production functions. The subroutine OWNOUT currently on the disk must be replaced by one of your own design if this option is to be utilized. Exhibit II.1 shows the control cards necessary to do this and specifies the structure and function of an OWNOUT subroutine.

It should be noted that the GENR and OUT links in TSP can also be used to create new variables.

If ITC is zero, after all information about a particular trial has been printed, the program will type "TO PUNCH, TURN SS2 ON BEFORE PUSHING START", and it will pause. To cause the data just generated to be punched, you must turn sense switch 2 ON and press the START button. If sense switch 2 is OFF when START is pushed, no deck will be created. If ITC is a number other than zero, the program will not request a decision from the operator and will not pause. No data will

punched; the program will proceed to the next trial.

This feature is useful for initial experiments when a large number of trials are to be made. Its use is essential when the leader of the exercise cannot be present when GBEAST is being run.

Each coefficient card must be followed by NT trial cards punched as follows:

GBEAST Trial Card

Columns	Variable	Explanation
1-3	NOB	The number of observations to be generated. Must be less than 200. Right-justified, no decimal point.
4-13	$X_1(1)$	The value of X_1 for the first observation.
14-23	$X_2(1)$	The value of X_2 for the first observation.
24-29	AG_1	The average percentage growth of X_1 per period.
30-35	AG_2	The average percentage growth of X_2 per period.
36-41	SG_1	The standard deviation of the percentage rate of growth of X_1.
42-47	SG_2	The standard deviation of the percentage rate of growth of X_2.
48-57	SDE	The standard deviation of the error term u.
68-65	SEED	(Optional) See text.

Exhibit II.1

A SAMPLE OWNOUT ROUTINE

```
++   DUP
 *DELETØWNØUT
++   FØR
 *LDISKØWNØUT
       SUBRØUTINE ØWNØUT (N, X)
       DIMENSIØN  X (3,200)
     C   PERFØRM ALL TRANSFØRMATIØNS AND DØ ALL PUNCHING HERE.
     C   YØU MUST PRØVIDE FØRMAT AND PUNCH STATEMENTS.
     C   ØWNØUT WILL ØNLY BE CALLED ØNCE BY THE MAIN PRØGRAM.
     C   SØ YØU MUST PUNCH ALL DATA BEFØRE RETURNING.
     C   N IS THE NUMBER ØF ØBSERVATIØNS,
     C   X(1,I)  =  Y(I),  X(2,I)  =  X1(I), X(3,I)  = X2(I).
       RETURN
       END
```

GBEAST generates values of X_1 and X_2 for observations 2 through NOB according to

(II.1) $X_i(t+1) = X_i(t)\ G_i(t)$ $i=1,2$ $t=1,\ldots,NOB-1.$

$G_i(t)$ is a normally distributed random variable with mean $(1 + AG_i/100)$ and standard deviation $(SG_i/100)$. Thus the AG's and the SG's should be punched as percentages - without the % sign, of course.

By varying these four parameters, any desired combination of trend and noise can be built into the X's.

Once the X's corresponding to a particular observation have been generated, the program calculates Y. GRNUM produces a value for u with mean zero and standard deviation SDE. These three numbers, along with the coefficients specified, determine the value of Y. See Appendix B.

If a value for SEED is specified, a particular sequence of G's and u's can be re-created. If columns 58-65 are left blank, the program will supply a SEED and thus will itself choose the "random" numbers generated. A SEED may be any five digits followed by a decimal point. See Appendix A for more details.

Exhibit II.2 shows a sample input deck to GBEAST.

Output

Each trial produces a separate page of output. A sample page is shown as Exhibit II.3.

The first few lines give basic identifying information: the title,

coefficients and trial number. The natural logarithm of the coefficient a is given, since that is what will be estimated by a regression on the generated data. The number of observations to be generated by this trial is also printed.

The next paragraph summarizes the information given on the trial card. This should help in producing reasonable data; it should also serve as a check against keypunch errors. If no SEED was specified, the value listed is the one supplied by the program.

The block headed by "OUTPUT" gives the means and standard deviations of the three variables generated and of their logarithms. These are unbiased estimates. That is, division is by (NOB - 1). Also, the standard deviation of the generated values of u is calculated. This is the true standard deviation of the error term.

The only way to see for sure if a particular set of data will yield a good fit and reasonable coefficient estimates is to test it on TSP or some other regression package. The next paragraph of output is an attempt to tell in rough terms what such a test will yield.[2]

The value for R-SQUARE is calculated as

(II.2) R-SQUARE = 1 -(Variance (u)/Variance(log y)).

---

[2] It would be possible, of course, to perform a regression within GBEAST itself. It was felt, however, that adding this feature would take more programming time and would involve more computation per trial than it was worth. The ambitious user with lots of computer time should be able to build in a regression routine of his own, as all the required numbers are present.

EXHIBIT II. 2

SAMPLE INPUT TO GBEAST

```
✦ ✦XEQSGBEAST
  R SCHMALENSEE - JULY, 1968 - TEST INPUT
  5 1       1.00       .75        .25
  20        20.        20.   1.0   0.0      1.5      0.0      .01
  20        20.        20.   1.0   1.0       .5      0.5      .02
  20        20.        20.   1.0   2.0       .5      1.0      .03
  20        20.        20.   1.0   3.0       .5      1.5      .04
  20        20.        20.   1.0   4.0       .5      2.0      .05
  3         5.00       .75        .25
  20        25.        25.   2.0   1.0      2.0      4.0               .01
  20        25.        25.   2.0   4.0      2.0      1.0               .01
  20        25.        25.   3.0   3.0      3.0      3.0               .01
                                              (CØLUMNS 1 - 35 BLANK)
```

EXHIBIT II. 3

SAMPLE OUTPUT FROM GBEAST

R SCHMALENSEE - JULY, 1968 - TEST INPUT

COEFFICIENTS (A, B1, B2) = (5.0000. .7500. .2500)
LOG(A) = 1.6094
TRIAL NUMBER 3 NUMBER OF OBSERVATIONS 20

INPUT	X1	X2
INITIAL VALUES	25.0000	25.0000
AVG PCT GROWTH	3.0000	3.0000
STD DEV OF PCT GROWTH	3.0000	3.0000
STD DEV OF ERROR (LOGS)	.0100	

SEED WAS 5452.

OUTPUT	Y	X1	X2
SAMPLE MEAN	163.3068	32.5373	33.6661
SAMPLE MEAN (LOGS)	5.0834	3.4734	3.4890
SAMPLE STD DEV	25.8903	4.3787	8.1977
SAMPLE STD DEV (LOGS)	.1607	.1384	.2387
STD DEV OF ERROR (LOGS)	.0104		

APPROX STATISTICS

R- SQUARE .995

TWO SIGMA INTERVALS (STD ERR)

LOG(A)	1.4281,	1.7907	(.0906)
B1	.6446,	.8553	(.0526)
B2	.1889,	.3110	(.0305)

DET(XTX) .8457

DECK NOT CREATED

The next set of numbers gives standard errors and 95% confidence intervals on the estimates of the coefficients. Since the true values of the coefficients and the error terms are known, it is possible to state that 95% of the time the estimates of these coefficients will lie in the interval shown. To put it another way, the estimated coefficients are random variables, normally distributed with means equal to the true values and standard deviations equal to the indicated standard errors.

The last statistic in this paragraph is an index of the reliability of the above-mentioned numbers. As is well known, the standard error of a regression coefficient is given by (II.3) $SE_i = \sigma_u \sqrt{a_{ii}}$, where σ_u is the standard deviation of the error, and a_{ii} is the i^{th} diagonal element of the inverse of (X^tX) where X^t is the transpose of X. The matrix X has as its i^{th} row the values of the independent variables corresponding to the i^{th} observation.[3] GBEAST calculates the a_{ii}'s as the ratios of the cofactors of the diagonal elements in (X^tX) to the determinant of that matrix. If X_1 and X_2 are highly correlated or if their logarithms have very small standard deviations, (X^tX) will be nearly singular, and the calculation of its (small) determinant may be greatly influenced by round-off error.[4]

---

[3] See J. Johnston, Econometric Methods (McGraw-Hill, 1960), pages 115-135.

[4] What happens is that nearly equal numbers must be subtracted. For a discussion of this problem, see D. D. McCracken and W. S. Dorn, Numerical Methods and Fortran Programming (John Wiley, 1964), chapter 2.

So if the determinant is small, the calculated standard errors, and hence the confidence intervals, may be inaccurate. They should be of the right order of magnitude, however, in all but the most extreme cases. (If the calculated determinant is <u>negative</u>, the standard errors are of course meaningless.)

The last line of output indicates whether or not a deck has been punched. All decks created, whether by an OWNOUT routine or by GBEAST, are preceeded by an identity card, giving the coefficients used and the trial number.

Before a set of data is used in the BEAST, it should be tested on a regression package to see what the actual coefficient estimates are. Assuming OWNOUT was not used, Exhibit (II.4) shows the makeup of a deck to perform such a test on TSP. The letters "NOB" should, of course, be replaced by the actual number of observations.

The use of generated data in sub-tasks 1 and 2 of BEAST will be discussed in Chapter IV.

EXHIBIT II.4

A TSP TEST DECK

```
+ + TSP
TØ TEST GBEAST DATA
SETL NØB
LØAD
LØAD ID X Y Z 1 NØB
ID
X
Y
Z
*****      INSERT A DATA DECK FRØM GBEAST   *****
END
GENR
XL = LØG (X)
LØG X
YL =LØG(X)
LØG  Y
ZL=LØG(Z)
LØG Z
END
REGR
SMPL 1 NØB
ØLSQ XL C YL ZL
END
KILL
```

III

The SBEAST Program

GBEAST is designed to make it as easy as possible to generate "reasonable" data for sub-tasks 1 and 2 of the BEAST. Similarly, SBEAST is designed to aid the instructor in providing "reasonable" values for income (Y), the number of firms (n), machines per firm (K), the wage rate (w), and the rental or per-period cost of machines (r) for sub-tasks 3 - 5 of the BEAST. SBEAST solves sub-tasks 3 and 4 and indicates how those solutions would vary as Y, n, K, and w are varied. It calculates the marginal revenue product of capital, given which sub-task 5 is trivial. SBEAST enables you to link the demand and production functions so that equilibrium price (P) and quantity (Q) are not too far removed from the sample means of those variables.

Again, we will discuss first the input to SBEAST and then the output it produces. As in chapter II, we will assume that SBEAST has been loaded onto the disk as a main program. Appendix C is a listing of SBEAST.

Input

The first three cards in an input deck correspond to the first three in a GBEAST deck: a job card, an * *XEOSSBEAST card, and a comment card. The last card in the deck must have columns 1 -35 blank.

In between, there may be one or more coefficient decks. We will

write the industry demand function to be examined as

$$\text{(III.1)} \quad Q = a\, Y^{b}\, P^{c},$$

and the firm production function as

$$\text{(III.2)} \quad Q = A\, L^{B}\, K^{C}.$$

The first card in a coefficient deck is a coefficient card, punched as follows:

SBEAST Coefficient Card

Columns	Variable	Explanation
1-2	NT	The number of trials to be made with these coefficients. Right-justified, no decimal point.
3-12	a	Coefficients of the industry demand function, as shown in equation (III.1).
13-22	b	
23-32	c	
33-42	A	Coefficients of the firm production, as shown in equation (III.2).
43-52	B	
53-62	C	

Each coefficient card must be followed by NT trial cards punched as follows:

SBEAST Trial Card

Columns	Variable	Explanation
1-15	Y	Income per consumer - eqn. (III.1).
16-30	n	The number of producing firms.
31-45	K	Machines per firm - eqn. (III.2).
46-60	w	The wage rate (per manhour)

Sample input to SBEAST is shown as Exhibit III.1.

Output

Sample output from SBEAST is shown as Exhibit III.2. The first line is, of course, your title or comment. If n is equal to one, SBEAST assumes a monopoly and solves sub-task 4 of the BEAST. The word "MONOPOLY" will then be the second line output. If n is greater than one, competition is assumed, "COMPETITION" is printed, and sub-task 3 is solved.

The next several lines list the information given on the coefficient and trial cards. This should serve both as a design aid and as a check against keypunch errors.

The next section presents the equilibrium values of price and quantity for the market, and of man-hours employed by each firm. It also indicates how the values of P and Q will vary from trial to trial.

These computations amount essentially to solving three linear equations in three unknowns. Equations (III.1) and (III.2) along with the value of N yield.

(III.3) $\ln(Q) = \ln(a) + b\ \ln(Y) + c\ \ln(P)$, and

(III.4) $\ln(Q) = \ln(n) + \ln(A) + B\ \ln(L) + C\ \ln(K)$.

The third equation comes from the requirement that labor be hired up to the point that its marginal revenue product equals the wage rate. Under competition, this implies

(III.5a) $\ln(L) = \ln(P) + [\ln(Q) - \ln(n)] + \ln(B) - \ln(w)$,

Exhibit III.1

<u>Sample input to SBEAST</u>

```
+ + XEQSSBEAST
  R SCHMALENSEE - July, 1968 - TEST INPUT
  4       1.0       1.0    -2.0        1.0       .5          .5
           1000.           10.            50.              1.0
           1000.           1.0            500.             1.0
           1000.           10.            50.              .5
           1000.           1.0            500.             .5

3  2.00       .95      -2.13      2.10      .524      .617
1500.            20.            62.            1.5
1500.            1.0            500.           1.0
1500.            20.            62.            .5

                                (CØLUMNS 1 - 35 BLANK)
```

Exhibit III.2

Sample Output from SBEAST

R SCHMALENSEE - JULY, 1968 - TEST INPUT

COMPETITION

INPUT

COEFFICIENTS

DEMAND	1.00,	1.00,	-2.00
PRODUCTION	1.00,	.50,	.50

TRIAL NUMBER 1

INCOME (Y) =	1000.0000
NO. OF FIRMS (N) =	10.0000
MACHINES/FIRM (K) =	50.0000
WAGE RATE (W) =	1.0000

OUTPUT

PRICE =	1.5874
OUTPUT =	396.8501
LABOR/FIRM =	31.4980

ELASTICITIES	PRICE	QUANTITY
Y	.3333	.3333
N	-.3333	.6666
K	-.3333	.6666
W	.3333	-.6666

DERIVITIVES	PRICE	QUANTITY
Y	.0005	.1322
N	-.0529	26.4566
K	-.0105	5.2913
W	.5291	-264.5667

MARG REV PROD OF K = .6299

since all firms will produce the same amount. Under monopoly (n = 1), we must consider marginal revenue instead of price; this provides

$$\text{(III.5b)} \qquad \ln(L) = \ln(P) + \ln(1+1/c) + \ln(B) - \ln(w).$$

If the price elasticity of demand, c, is greater than or equal to minus one, there is no equilibrium monopoly output. Profits rise with P everywhere. Should you specify $c \geq -1$ and N = 1, SBEAST will inform you of your error and will go on to the next trial.

The solution of (III.3) - (III.5) for ln(P) and ln(Q) yields

$$\text{(III.6)} \qquad \ln(P) = PC_1 + PC_2 \ln(Y) + PC_3 \ln(n) + PC_4 \ln(K) + PC_5 \ln(w), \text{ and}$$

$$\text{(III.7)} \qquad \ln(Q) = QC_1 + QC_2 \ln(Y) + QC_3 \ln(n) + QC_4 \ln(K) + QC_5 \ln(w)$$

All coefficients except PC_1 and QC_1 are the same under monopoly as under competition. If we let

$$D = B - c + cB,$$

these coefficients are as follows:

$$PC_2 = \frac{b(1-B)}{D} \qquad QC_2 = \frac{bB}{D}$$

$$PC_3 = \frac{\text{[illegible]}-1)}{D} \qquad QC_3 = \frac{c(B-1)}{D}$$

$$PC_4 = \frac{-C}{D} \qquad QC_4 = \frac{-cC}{D}$$

$$PC_5 = \frac{B}{D} \qquad QC_5 = \frac{cB}{D}$$

Under competition, we have

$$PC_1 = \frac{(1-B)\ln(a)-\ln(A) - B\ln(B)}{D}$$

$$QC_1 = \frac{B\ln(a) - c\ln(A) - cB\ln(B),}{D}$$

and under monopoly,

$$PC_1 = \frac{(1-B)\ln(a) - \ln(A) - B\ln(B) + B\ln[c/(c+1)]}{D}$$

$$QC_1 = \frac{B\ln(a) - c\ln(A) - cB\ln(B) + cB\ln[c/(c+1)]}{D}.$$

Labor per firm is then obtained from (III.5a) or (III.5b).

Clearly PC_2 - PC_5 and QC_2 - QC_5 are the elasticities shown in the output.[5] The partial derivitives are obtained from these elasticities by multiplying by either P or Q and dividing by the independent variable specified. Both the elasticities and the derivitives are intended to aid you in obtaining reasonable values for P, Q, and L.

The marginal revenue product of capital is calculated from

$$\text{(III.8a)} \qquad r^* = \frac{PQC}{nK}$$

---

[5]For the monopoly case, derivitives and elasticities with respect to n are meaningless. Hence they are not given as output.

under competititon[6] and

$$\text{(III.8b)} \qquad r^* = \frac{P(c-1)QC}{cK}$$

under monopoly. If, in problem 5, you specify an r less than r*, it is in the interest of the firm or firms involved to expand. If, on the other hand, the market price of machine services is above their marginal revenue product, they should contract capacity.

We will now put these pieces together and discuss the BEAST as a whole.

---

[6] Recall that each firm will produce the same amount, Q/n.

under competition or under monopoly, [illegible]

(III.8a) [illegible]

under monopoly. If, in problem 3), you specify [illegible] less than [illegible], it is in the interest of the firm or firms involved to expand. If, on the other hand, the market price of hauling services is above their marginal revenue product, they should contract capacity.

We will now put these pieces together and discuss the model as a whole.

IV

Care and Feeding

The programs GBEAST and SBEAST (and TSP) are tools to be used by the instructor in putting together a particular version of the BEAST. In Chapters II and III, we discussed the mechanics of using them. We now turn to the broader issues of what a BEAST should look like and how it should be administered.

Design

It must be emphasized that all data for sub-tasks 1 and must be generated and the values to be used in sub-tasks 3-5 should be specified before students begin work. Experience suggests that students will have a multitude of mechanical and conceptual problems with the BEAST and that adequately handling their questions precludes further design work.

It is more important in sub-tasks 1 and 2 that the students obtain reasonable parameter estimates than that they get a good R^2. Since the forecasts to be made in sub-tasks 3-5 are of points on the true functions, the accuracy of their results depends only on the quality of their coefficient estimates. Of course, all estimated parameters should be statistically significant; they should be at least twice as large as their (estimated) standard errors.

Most students will fit a linear relation before trying one linear in the logarithms. To simplify the calculations in sub-tasks 3-5 and to

ensure satisfying predictions, the linear function should not fit as well as the "correct" logarithmic one. This can be brought about by having the independent variables [log(X_1) and log(X_2)] vary over a fairly wide range, so the real non-linearity makes itself felt.[7] Note that a large value for the standard deviation does not necessarily indicate a wide range, as it can be caused by either a large AG or a large SG. Only the former will ensure a wide range. A quick estimate of the range can be made by comparing the sample mean with the initial value.

It can be argued that the linear relation need not fit worse for the production function data. You may well feel that any student who does not notice that the linear form is generally inconsistent with the firm's hiring both capital and labor deserves to be misled.

The linear relation should be compared to the log-linear form. Thus a complete TSP test deck would include one card not shown in Exhibit II.4. Directly above or below the OLSQ card shown there, place a card punched

OLSQ X C Y Z.

There are thus three broad tests that should be met by a good data deck:

1. The estimated coefficients should be close to their true

---

[7] Wide ranges of the X's and Y also simplify the design of subtasks 3-5 as we shall see below.

enough satisfying predictions, the linear function should not fit as well as the "correct" logarithmic one. This can be brought about by having the independent variables (i.e., Weber ratios) vary over a fairly wide range, so the test conditions were set up (pp. 18ff.). [illegible] values and should be more than twice their estimated standard errors.

2. The ranges of $\log(X_1)$ and $\log(X_2)$ (and thus $\log(Y)$) should be large.

3. The linear form should not fit as well as (should have a lower R^2 than) the log-linear relation.

From the point of view of design, sub-tasks 1 and 2 are not independent. They are tied together by the rest of the BEAST. Since one will normally choose a fairly large n for sub-task 3, it is generally desirable to have the mean value of quantity demanded in the market a good deal larger than the mean value of one firm's production. If sub-task 4 is more important in the designer's eyes, they should have nearly equal means.

Again the object is to give reasonably good predictions. The nearer the forecast variables involved in sub-tasks 3 and 4 are to their sample means, the better will be the forecasts made. Any regression line will pass through the sample means; the farther away you go, the more important errors in the coefficient estimates become.[8]

Suppose now that GBEAST and TSP have been employed, and the two data decks and their actual and estimated coefficients are in hand. It is now time to use SBEAST.

---

[8] C. f. Johnston, op. cit., pp. 34-39.

The first coefficient card will generally contain the true values. The test values for Y and K should generally be within the sample range, as argued above. If this is to be done for both sub-tasks 3 and 4, wide ranges will be helpful. Different values of n and w should be tried to link the two functions. The equilibrium values of P, Q, and L should again be somewhere within the sample range.

Once reasonable values for Y, L, K, and w for sub-tasks 3 and 4 have been found using the true coefficients, these values should be run using the coefficients that the students will estimate. This provides an easy check on their arithmetic. Also, if the students' results seem too far away from the true figures, you may wish to experiment further to get P and Q closer to their sample means.

Once sub-tasks 3 and 4 have been constructed, sub-task 5 merely involves selecting a value for the rental on machines and comparing it with the marginal product of capital as calculated by SBEAST. It should not be too hard to ensure that the studnets will get the right answer.

Administration

It is suggested that three papers be required of the students:

1. A short paper on the demand estimates. At least two functional forms must be compared on statistical and economic grounds, and the preferred form indicated. This should require no more than 2-3 pages.

2. A similar short paper on the production function estimates.

3. A short paper presenting the methods used and the results obtained for whichever of sub-tasks 3-5 are assigned.

Titled computer output (see below) should be required to be attached to the first two papers, to ensure that the student has actually run the regressions.

The reasons for two papers on the estimations is that linear regression does not come that easily to most students. They will make many errors on the first papers. Careful grading can provide valuable feedback and can enable them to produce second papers which will satisfy both you and them.

Arithmetic is obviously of secondary importance in sub-tasks 3-5. You may wish to see it as an appendix to the third paper, but there seems no reason to impose on the students the burden of preparing it neatly.

Now the mechanics. All students should be given a TSP writeup when linear regression is discussed in class. Indeed, the author has found that a page of TSP output is a very good device to use in presenting and discussing the important statistics provided by a typical regression package.

Sub-tasks 1 and 2 are administered identically. The students should first be given a listing of the demand (production) function data and an instruction sheet. This should

1. Tell them that they are to estimate a demand (production)

function for use in a forecasting problem.

2. Describe the list of data.
3. Tell them when the paper is due, how long it is to be, and what it is to contain.
4. Spell out the mechanics of the exercises, including the deadlines for deck submittal and the makeup of each deck.

Under (2), each series should be given a one-letter name. Under (4), tell them that these series will be loaded under those names. They may then submit a GENR phase to perform transformations and construct new variables. Any new variables they construct should be given names consisting of four letters, the last two of which are their initials. This will minimize the chances of two students generating a variable with the same name.

They must, naturally, have a REGR phase. To conserve computer time, a limit should be placed on the number of regressions per student and per run. To be able to separate output, each OLSQ card should be preceeded with a RMRK card and a card with the student's name punched in columns 2-80. This will cause the student's name to be printed on each page of his output. It will then be possible to tell which student have in fact made runs.

Finally, more than one run should be scheduled for each of sub-tasks 1 and 2. Even the best students make keypunch errors.

The instructor then signs up for a large block of computer time and prepares a deck consisting of the usual TSP header cards plus

a MODE EXPR card plus a LOAD link. The students' decks follow the LOAD link and are themselves followed by a KILL card. Exhibit IV.1 indicates the makeup of this deck.

The MODE EXPR feature is quite essential. When such a card has been read in, TSP will pause when an error has been encountered. Pressing the START button will cause it to ignore the offending card and attempt to go on. Generally this will be possible, and student errors will not result in a program stop. When they do, it will be necessary to remove the offending student deck and to start TSP over again.

The above procedure is designed to minimize the burden on the computer facility. With a small class, it may be simpler for all concerned to have the students submit their decks directly. They must then be provided with data decks to load themselves. All regressions should still be titled, of course.

Grading of sub-tasks 1 and 2 should emphasize understanding of the statistical and economic issues, not results.

Sub-tasks 3-5 are easier for the instructor, but may well be more difficult for the students. Give them plenty of time to do the calculations. Your instruction sheet should tell them

1. The forecasts to be made, and the relevant values of Y, n, K, w, and r for each one.

2. How long their paper should be, its general form, and when it is due.

Again, they should be required to present methods and answers, not neat arithmetic. When they say "marginal revenue", for instance, they should make it clear how it was calculated, not what numerical values it had.

The students will want to know what the "correct" answers to these tasks are. They should be told.

Exhibit IV.1

A Deck to Run Students' Regressions

```
✦ ✦ TSP
 TITLE IN CØLUMNS 2 - 80
 SETL NØB
 MØDE EXPR
 LØAD
 LØAD ID X Y Z 1 NØB
 * INSERT THE DATA DECK FRØM GBEAST HERE *
 END

        *
        *
        *

  INSERT STUDENT DECKS HERE.  THEY SHØULD BE ØF THE FØLLØWING FØRM.

GENR
 * DESIRED TRANSFØRMATIØNS *
END
REGR
 * DESIRED REGRESSIØNS *
END
        *
        *
        *

KILL
```

Appendix A

Description and Listing of GRNUM*

GRNUM is the subrountine utilized by GBEAST to generate normally distributed random numbers with specified mean and standard deviation. These numbers are used to produce successive values of X_1, X_2, and the error term u. This Appendix will briefly explain how GRNUM performs this function.

GRNUM receives from the calling program values for CT, SD, and S. It then (1) generates two random numbers uniformly distributed between zero and one and (2) transforms them into a single normally distributed random number with mean CT and standard deviation SD. This number is returned to the calling program as RN. A new value for S is also returned. We will discuss these operations in turn.

The uniformly distributed random numbers are generated by the multiplicitive congrunetial method. That is, a sequence of X's is generated according to

(A.1) $X_{n+1} \equiv A\ X_n \pmod{M}$**

---

*This Appendix and GRNUM borrow heavily from A. K. Dixit, "Stochastic Simulation in Econometrics: A First Report", Econometricks Memo #2, Department of Economics, MIT, July, 1967.

**These expression $X \equiv Y \pmod{M}$ means that (Y-X) is exactly divisible by M. In equation (A.1), X_{n+1} is equal to M times the fractional part of $(A\ X_n/M)$. For example, 1 = 8 (mod 7), 5 = 15 (mod 10). GRNUM uses $M \overset{n}{=} 10^5$, and we have 83542. = 7583542. (mod 10^5).

Then the sequence (X_n/M) is used as independent random numbers uniformly distributed over the interval zero to one. We generate a deterministic sequence that "looks like" a set of random numbers.

Here $M = 10^5$, and X_0 is the value of S taken from the calling program and referred to as SEED in Chapter II. The value computed as X_2 is returned as the new value for S. The value of A chosen yields low serial correlation in the X's and gives that sequence the maximum period (5,000) possible with $M = 10^5$.

The two uniformly distributed "random" numbers Y_1 $(=X_1/M)$ and Y_2 $(=X_2/M)$ are transformed into a normally distributed random number according to an exact transformation suggested by Box and Mueller***. To obtain two independent normal deviates with mean zero and standard deviation one, we can calculate

(A.2)
$$Z_1 = \sqrt{-2\ln(Y_1)} \quad \cos(2\ Y_2)$$
$$Z_2 = \sqrt{-2\ln(Y_1)} \quad \sin(2\ Y_2)$$

Then $(Z_1 + Z_2)$ will have mean zero and standard deviation $\sqrt{2}$, and

(A.3) $$RN = CT + (Z_1+Z_2)\ SD/\sqrt{2}$$

will be normal with mean CT and standard deviation SD, as required. Successive observation of RN will be independent.

As a test, 500 successive (different) values of RN were generated by GRNUM. These had a first order serial correlation coefficient of .05. A chi-square test for goodness of fit to the normal dis-

***G.E.P. Box and M. E. Miller, "A Note on the Generation of Normal Deviates", Ann. Math, Stat., 29, (1958), pp. 610-11. Discussed in Dixit, op. cit.

tribution was performed, with 7 degrees of freedom. The value of chi-square was less than 5, indicating quite a good fit. Inspection of the frequency distribution confirmed this.

A listing of GRNUM follows.

A Listing of GRNUM

```
      SUBRØUTINE GRNUM (RN, CT, SD, S)
      DIMENSIØN V(2)
      A = 283.
      DØ 01 1=1,2
      F=A*S/10.**5
      K=F
      FK=K
      V(1)=F-FK
01    S=V(1)*10.**5
      A=(-LØGF(V(1)))**.5
      R=6.2831853*V(2)
      B=CØSF(R)+SINF(R)
      RN=CT+SD*A*B
      RETURN
      END
```

APPENDIX B - A LISTING ØF GBEAST

```
1000 FØRMAT(1X,A3,19A4)
1001 FØRMAT(I2,2I1,3F10.2)
1002  FØRMAT(I3,2F10.2,4F6.2,F10.2,F8.0)
1003 FØRMAT(1H1,A3,19A4)
1004 FØRMAT(1H0,26HCØEFFICIENTS (A,B1,B2) = (, F15.4,1H,,F15.4,1H,,F15.
    14,1H))
1005 FØRMAT(1X,13HTRIAL NUMBER ,I2,5X,23HNUMBER ØF ØBSERVATIØNS ,I3)
1006 FØRMAT(1H0,9X,5HINPUT)
1007 FØRMAT(1X ,30X,2HX1,13X,2HX2)
1008 FØRMAT(1X,14HINITIAL VALUES,7X,2F15.4)
1009 FØRMAT(1X,14HAVG PCT GRØWTH,7X,2F15.4)
1010 FØRMAT(1X,21HSTD DEV ØF PCT GRØWTH,2F15.4)
1011 FØRMAT(1H0,23HSTD DEV ØF ERRØR (LØGS),F13.4)
1012 FØRMAT(1X,9HSEED WAS ,F7.0)
1013 FØRMAT(1H0,9X,6HØUTPUT)
1014 FØRMAT(2X ,29X,1HY,14X,2HX1,13X,2HX2)
1015 FØRMAT(1X,11HSAMPLE MEAN,10X,3F15.4)
1016 FØRMAT(1X,14HSAMPLE STD DEV,7X,3F15.4),
1017 FØRMAT(1X,21HSAMPLE STD DEV (LØGS),3F15.4)
1018 FØRMAT(1H0,9X,17HAPPRØX STATISTICS)
1019 FØRMAT(1H0,8HR-SQUARE,F6.3)
1020 FØRMAT(1H0,29HTWØ SIGMA INTERVALS (STD ERR))
1021 FØRMAT(5X,6HLØG(A),F15.4,1H,,F15.4,5X,1H(,F15.4,1H))
1022 FØRMAT(7X,2HB1,F17.4,1H,,F15.4,5X,1H(,F15.4,1H))
1023 FØRMAT(7X,2HB2,F17.4,1H,,F15.4,5X,1H(,F15.4,1H))
1024 FØRMAT(1X ,42HTØ PUNCH, TURN SS2 ØN BEFØRE PUSHING START)
1025 FØRMAT(1X,16HDECK NØT CREATED)
1026 FØRMAT(1X,12HDECK CREATED)
1027 FØRMAT(1H )
```

```
1028 FØRMAT(1X,18HSAMPLE MEAN (LØGS),5X,3F15.4)
1029 FØRMAT(1X,I3,3F15.4)
1030 FØRMAT(5X,12HCØEFFICIENTS,3F15.4,5X,13HTRIAL NUMBER ,I2)
1031 FØRMAT(1X,8HLØG(A)* =,F15.4)
1032 FØRMAT(1HO,8HDET(XTX),F15.4)
     DIMENSIØN TITLE(20),AG(2),SG(2),SM(3),SMSQ(3),SMA(3),SMSQA(3),XL(3
    1),X(3,200),FQA(3),FQ(3)
C                        READ CØNTRØL CARDS - BLANK KILLS            ********
     SED=54732.
     READ 1000, (TITLE(I),I=1,20)
01   READ 1001,NDK,IØC,ITC,A,B,C
     IF(NDK) 02,99,02
02   DØ 88 KK=1,NDK
     READ 1002,NØB,X(2,1),X(3,1),AG(1),AG(2),SG(1),SG(2),SE,SDN
     IF(SDN) 04,04,03
03   SED=SDN
C                        PRINT INPUT INFØ                            ********
04   PRINT 1003,(TITLE(I),I=1,20)
     PRINT 1004,A,B,C
     ALG=LØGF(A)
     PRINT 1031,ALG
     PRINT 1005,KK,NØB
     PRINT 1027
     PRINT 1006
     PRINT 1007
     PRINT 1008,X(2,1),X(3,1)
     PRINT 1009,AG(1),AG(2)
     PRINT 1010,SG(1),SG(2)
     PRINT 1011,SE
     PRINT 1012, SED
C                        INITIALIZE                                  ********
     DØ 77 I=1,2
     AG(I)=1.+AG(I)/100.
77   SG(I)=SG(I)/100.
     DØ 05 I=1,3
     SM(I)=0.
     SMSQ(I)=0.
     SMA(I)=0.
05   SMSQA(I)=0.
     SME=0.
```

```
      SMSQE=0.
      SCP=0.
C                       GENERATE DATA AND RELEVANT SUMS              ********
      DØ 11 I=1,NØB
      IF(I-1) 07,09,07
   07 DØ 08 J=1,2
      CALL GRNUM (G,AG(J),SG(J),SED)
   08 X(J+1,I)=X(J+1,I-1)*G
   09 CALL GRNUM(E, 0.0, SE, SED)
      X(1,I)=A*(X(2,I)**B)*(X(3,I)**C)*EXPF(E)
      DØ 10 J=1,3
      SMA(J)=SMA(J)+X(J,I)
      SMSQA(J)=SMSQA(J)+X(J,I)**2
      XL(J)=LØGF(X(J,I))
      SM(J)=SM(J)+XL(J)
   10 SMSQ(J)=SMSQ(J)+XL(J)**2
      SME=SME+E
      SMSQE=SMSQE+E**2
   11 SCP=SCP+XL(2)*XL(3)
C                       GENERATE AND PRINT OUTPUT INFO               **********
      PRINT 1027
      PRINT 1013
      PRINT 1014
      FN=NØB
      SDE=SQRTF((SMSQE-SME**2/FN)/(FN-1.))
      CF1=FN*(SMSQ(2)*SMSQ(3)-SCP**2)
      CF2=-SM(2)*(SM(2)*SMSQ(3)-SCP*SM(3))
      CF3=SM(3)*(SM(2)*SCP-SM(3)*SMSQ(2))
      D=CF1+CF2+CF3
      CFSE=SDE*FN/SQRTF(D)
      DENA=SQRTF(SMSQ(2)*SMSQ(3)-SCP**2)/FN
      DØ 12 J=1,3
      SMSQA(J)=SQRTF(SMSQA(J)/FN- (SMA(J)**2)/(FN**2))
      SMA(J)=SMA(J)/FN
      SMSQ(J)=SQRTF(SMSQ(J)/FN - (SM(J)**2)/(FN**2))
   12 SM(J)=SM(J)/FN
      PRINT 1015,(SMA(J),J=1,3)
      PRINT 1028,(SM(J),J=1,3)
      CFAC=SQRTF(FN/(FN-1.))
      DØ 69 J=1,3
```

```
      FQA(J)=SMSQA(J)*CFAC
   69 FQ(J)=SMSQ(J)*CFAC
      PRINT 1016,(  FQA(J),J=1,3)
      PRINT 1017,(  FQ(J),J=1,3)
      PRINT 1011,SDE
C                    GENERATE AND PRINT APPROX STATISTICS          ********
      PRINT 1027
      PRINT 1018
      RS = 1. - (SDE**2)/(FQ(1)**2)
      PRINT 1019,RS
      SEA=CFSE*DENA
      SEB=CFSE*SMSQ(3)
      SEC=CFSE*SMSQ(2)
      SWA=SEA*2.
      SWB=SEB*2.
      SWC=SEC*2.
      PRINT 1020
      AL=ALG-SWA
      AU=ALG+SWA
      PRINT 1021,AL,AU,SEA
      BL=B-SWB
      BU=B+SWB
      PRINT 1022,BL,BU,SEB
      CL=C-SWC
      CU=C+SWC
      PRINT 1023,CL,CU,SEC
      PRINT 1032,D
C                    PUNCH DECK IF REQUIRED - CHECK FØR ØWNØUT     ********
      IF(ITC) 17,18,17
   18 PRINT 1027
      TYPE 1024
      PAUSE
      IF(SENSE SWITCH 2) 13,17
   13 PRINT 1026
      PUNCH 1030,A,B,C,NDK
      IF(IZC) 14,15,14
   14 CALL ØWNØUT(NØB,X)
      GØ TØ 88
   15 DØ 16 I=1,NØB
      PUNCH 1029,I,(X(J,I),J=1,3)
```

```
16    CØNTINUE
      GØ TØ 88
17    PRINT 1025
88    CØNTINUE
      GØ TØ 01
99    CALL EXIT
      END
```

APPENDIX C - A LISTING ØF SBEAST

```
1000 FØRMAT(1X,A3,19A4)
1001 FØRMAT(I2,6F10.2)
1002 FØRMAT(4F15.4)
1003 FØRMAT(1H1,A3,19A4)
1004 FØRMAT(1H0,6X,11HCØMPETITIØN)
1005 FØRMAT(1H0,7X,8HMØNØPØLY)
1006 FØRMAT(1H )
1007 FØRMAT(1H0,9X,5HINPUT)
1008 FØRMAT(1H0,12HCØEFFICIENTS)
1009 FØRMAT(5X,6HDEMAND,4X,F10.2,1H,,F10.2,1H,,F10.2)
1010 FØRMAT(5X,10HPRØDUCTIØN,F10.2,1H,,F10.2,1H,,F10.2)
1011 FØRMAT(1H0,13HTRIAL NUMBER ,I2)
1012 FØRMAT(5X,12HINCØME (Y) =,7X,F15.4)
1013 FØRMAT(5X,19HMACHINES/FIRM (K) =,F15.4)
1014 FØRMAT(5X,18HNØ. ØF FIRMS (N) =,1X,F15.4)
1015 FØRMAT(5X,15HWAGE RATE (W) =,4X,F15.4)
1016 FØRMAT(1H0,9X,6HØUTPUT)
1017 FØRMAT(1H0,4X,8HPRICE = ,F18.4)
1018 FØRMAT(5X,9HØUTPUT = ,F17.4)
1019 FØRMAT(1H0,12HELASTICITIES,5X,5HPRICE,7X,8HQUANTITY)
1020 FØRMAT(7X,1HY,2F15.4)
1021 FØRMAT(7X,1HN,2F15.4)
1022 FØRMAT(7X,1HK,2F15.4)
1023 FØRMAT(7X,1HW,2F15.4)
```

```
1024 FØRMAT(1H0,11HDERIVITIVES,6X,5HPRICE,7X,8HQUANTITY)
1025 FØRMAT(1H0,60HDEMAND IS INELASTIC, PRØFIT INCREASES WITH P AND HAS
    1 NØ MAX.)
1026 FØRMAT(1H0,20HMARG REV PRØD ØF K =,F15.4)
1027 FØRMAT(5X,12HLABØR/FIRM =,F15.4)
     DIMENSIØN TITLE(20),CDF(3),CPF(3),QC(5),PC(5)
C                         READ NUMBER ØF TRIALS, CØEFFS                ********
     READ 1000,(TITLE(I),I=1,20)
 01  READ 1001,NT,(CDF(I),I=1, 3),(CPF(I),I=1,3)
     IF(NT) 02,99,02
C                         CALCULATE ELASTICITIES - DEPENDS ØNLY ØN CØEFFS ********
 02  CDEN=CPF(2)-CDF(3)+CDF(3)*CPF(2)
     PC(2)=(CDF(2)*(1.-CPF(2)))/CDEN
     PC(3)=(CPF(2)-1.)/CDEN
     PC(4)=(-CPF(3))/CDEN
     PC(5)=(CPF(2))/CDEN
     QC(2)=(CDF(2)*CPF(2))/CDEN
     QC(3)=CDF(3)*(CPF(2)-1.)/CDEN
     QC(4)=(-CDF(3)*CPF(3))/CDEN
     QC(5) =(CDF(3)*CPF(2))/CDEN
C                         CALCULATE AND PRINT INFØ FØR EACH TRIAL      ********
 03  DØ 13 KK=1,NT
     READ 1002,Y,FN,FK,W
     IMT=FN-1.
C                         PRINT INPUT DESCRIPTIØN                      ********
     PRINT 1003,(TITLE(I),I=1,20)
     IF(IMT)05,04,05
 04  PRINT 1005
     GØ TØ 06
 05  PRINT 1004
 06  PRINT 1006
     PRINT 1007
     PRINT 1008
     PRINT 1009, (CDF(I),I=1,3)
     PRINT 1010, (CPF(I),I=1,3)
     PRINT 1011,KK
     PRINT 1012,Y
     PRINT 1014,FN
     PRINT 1013,FK
     PRINT 1015,W
```

```
C                       CALCULATE AND PRINT RESULTS                    ********
      PRINT 1036
      PRINT 1016
      PC(1)=EXPF(((1.-CPF(2))*LØGF(CDF(1))-LØGF(CPF(1))-CPF(2)*LØGF(CPF(
     12)))/CDEN)
      QC(1)=EXPF((CPF(2)*LØGF(CDF(1))-CDF(3)*(LØGF(CPF(1))+CPF(2)*LØGF(C
     1PF(2))))/CDEN)
      IF(IMT) 08,07,08
07    IF(CDF(3)+1.0) 27,17,17
17    PRINT 1025
      GØ TØ 13
27    F=CPF(2)*LØGF(CDF(3)/(CDF(3)+1.))/CDEN
      PC(1)=PC(1)*EXPF(F)
      QC(1)=QC(1)*EXPF(CDF(3)*F)
08    FP=PC(1)*(Y**PC(2))*(FN**PC(3))*FK**PC(4) *(W**PC(5))
      PRINT 1017,FP
      FQ=QC(1)*(Y**QC(2))*(FN**QC(3))*(FK**QC(4))*(W**QC(5))
      PRINT 1018,FQ
      FLF=(FP*FQ*CPF(2))/(W*FN)
      IF(IMT)71,70,71
70    FLF=FLF*(CDF(3)+1.)/CDF(3)
71    PRINT 1027,FLF
      PRINT 1019
      PRINT 1020,PC(2),QC(2)
      IF(IMT) 09,10,09
09    PRINT 1021,PC(3),QC(3)
10    PRINT 1022,PC(4),QC(4)
      PRINT 1023,PC(5),QC(5)
      PRINT 1024
      DA=PC(2)*FP/Y
      DB=QC(2)*FQ/Y
      PRINT 1020,DA,DB
      IF(IMT) 11,12,11
11    DA=PC(3)*FP/FN
      DB=QC(3)*FQ/FN
      PRINT 1021,DA,DB
12    DA=PC(4)*FP/FK
      DB=QC(4)*FQ/FK
      PRINT 1022,DA,DB
      DA=PC(5)*FP/W
```

```
      DB=QC(5)*FQ/W
      PRINT 1023,DA,DB
      FMRPK=(FP*FQ*CPF(3))/(FK*FN)
      IF(IMT) 15,14,15
14    FMRPK=FMRPK*(CDF(3)+1.)/CDF(3)
15    PRINT 1026,FMRPK
13    CØNTINUE
      GØ TØ 01
99    CALL EXIT
      END
```