AUTOSTAT: a Language for Statistical Data Processing

By A. S. Douglas* and A. J. Mitchell

A language is described in which it is possible to set up operations for statistical data processing in connection, for instance, with market-research surveys. The language is claimed to be comparatively easy to use, whilst at the same time comparatively straightforward to program for a iarge-scale electronic data-processing installation. Restrictions on the use of the language, when programmed for use with a Pegasus computer equipped with magnetic tape, are given. This paper was presented at the Second Conference of the British Computer Society in Harrogate on 5 July 1960.

1. Introduction

In order to carry out statistical analyses of large amounts of data it is first necessary to summarize the data under suitable headings. It is also essential to ensure the validity and correctness of the data collected. The latter is achieved by "editing" the raw data, that is by scanning it to detect omissions, mistakes and inconsistencies, before submitting it to a process of analysis.

The analysis of data is carried out by the use of a classification scheme, which depends both upon the nature of the data collected, and upon the use to which the analysis is to be put. Thus in setting up a questionnaire we may or may not include the question "What is your age?" If we do not include the question, the classification of informants by age cannot be attempted on the basis of answers to that questionnaire. However, the fact that this question is asked and answered does not imply that we actually require to use the answer in the form it is given. We may, in fact, only need to consult the answer in order to place the informant in a particular age grouping such as, for exampie, middle-age or teenage. In spite of the fact that it is not initially planned to use the full information as recorded, it may be convenient to collect the additional information in order to enable additional or amended classifications to be introduced later without necessitating the collection of further data.

Analysis normally comprises two processes, firstly the selection of data according to the classification scheme, and secondly, counting the number of items selected, possibly with varying weights attached to each item.

This selection and counting is often now carried out by the use of punched-card machines, called "counter-sorters," which have limited but useful facilities for sorting through required information and counting the number of items possessing a selected property.

When the desired counting has been completed, the results are normally used to prepare a suitable table of values for inspection. The figures so far obtained can also be used as the basis for statistical calculation, provided that a suitable number of sums have been made, not only of items and weighted items, but also of certain derivatives of these such as the squares of the weights.

* Dr. Douglas is now with C.E.I.R. (U.K.) Ltd., London, W.C.2.

The whole process can thus be considered as comprising three or four separate stages, namely

1. Editing
2. Selection and counting
3. Tabulation

and/or 4. Statistical analysis.

All of these functions can be carried out with an electronic computer. However, it is desirable that the specification of each stage should be as simple as possible for a user such as a Market Research Officer. Rather than attempt to train these users in computer programming it is obviously desirable to provide them with a suitable language in which to address the machinery to specify the required manipulations. *Autostat* is such a language.

2. The Language Processor

Introduction of a controlling language necessitates adding a fifth process to those described above. This comprises a language processing operation, the function of which is to translate the instructions of the user, given in Autostat language, into suitable parameters specifying the classification of the data, how it is to be stored in the computer, what editing procedures are to be applied, what counts are to be carried out, what statistical analyses are to be employed, and in what form the results are to be printed. This we shall designate Stage 0, Translation.

The details of the computer techniques used will be given in a later paper. Here we describe the language to be used to address the computer, and give a simple example of its application. This is not intended as a complete manual for its use, but outlines its main features.

3. Labelling of Data

We may suppose that the data to be prepared for input to a computer is in some arbitrary layout on a form arranged for the convenience, for instance, of an interviewer filling in the answers to a set of questions. The questions may be supposed to be numbered or labelled on the form in some way to identify the answers belonging to them, the same scheme being used on each form. For the purposes of input to the computer (Stage 1) we are not concerned directly with the informa-

tion content of the answers, but only with providing the computer with an adequate specification of the data which will be presented to it, and with the identification of unacceptable answers.

We allot to each item to be read in from the form an arbitrary label, called a "Q-label," which identifies the answer concerned. Specification of the method of preparation to be used is given to the machine in a series of statements about the Q-labels. For instance, we write

Q7, YES/NO.

This statement implies that the item to be labelled Q7 in the computer relates to the seventh item to be read in. This item can be either the set of symbols YES or the set NO; any other answer, or the lack of an answer, will be treated as an error in data preparation.

If we write Q7, YES/NO/–, then a distinction will be made between whether the question has been answered in either of these two ways or not answered at all. If it is not answered on the form, a dash must be inserted during punching.

To facilitate reading of answers with a long range of possible symbolic replies, we may also write, e.g.,

Q8, 1–99

when any two-digit combination will be accepted as a valid answer, and all such answers will be treated as alternative, or

Q9, 16–24/25–34/35–44/45–64/65–110,

when five different groups will be distinguished, an answer 37, for instance, falling in group 3. The actual numerical answer will not be recorded in this case unless the special symbol *n* is added to the statement. The function of this symbol will be discussed in more detail later.

It is permitted to mix letters and digits in the answer, and, in this case, the numbers and letters are supposed to be in ascending sequence as follows:

(no symbol), 0,1,2, . . ., 9,A,B,. . ., Z,

so that A2 > 2A > A > 2, for example.

Special provision is also made for the input of information recorded in pounds sterling. All values must be preceded by the symbol £, and shillings and pence must be entered as numbers; for example, we may write £54.0.0., but not £54.–.–. Shillings and pence must be separated by . and no other symbol. Deductions may be entered as, e.g., £—5.14.7. For the purposes of input specification, values may be treated in the same way as numerical data. We may write, for instance,

Q10, £0.0.0 — £9999.19.11.

to indicate that permissible answers to Q10 are positive, and lie in the range up to £10,000. We may also classify items by statements such as:

Q11, £0.0.0 — £49.19.11/£50.0.0.—
£99.19.11/£100.0.0 – £999.19.11.

In this case the actual value read in will not be recorded unless the special symbol *V* is added. This symbol has a function similar to that of *n* for numerical data, and will be further discussed with it later.

4. Regrouping the Data

In order to specify tabulations of the data, it is desirable both to relate the Q-labels to the information content of the answer recorded, and also, sometimes, to regroup the data. To facilitate this, provision is made to relabel data by the use of a combination of letters. This relabelling may be combined with a statement defining a Q-label, or may be separate from it. Thus we may write

AGE = Q9, 16–24/25–34/35–44/45–64/65–110;

this records the group number into which the age of the informant falls (the age being supposed to be the answer to Question 9 on the form) and at the same time relates that data to the label AGE. Alternatively we might write the two statements separately, thus:

Q9, 16–24/25–34/35–44/45–64/65–110
AGE = Q9.

The label AGE can then be used in the general sense "the answer to question 9." If we wish to refer during regrouping to all those informants in the age group 16–24, we may write Q9(1) or AGE (1). If there is no possibility of ambiguity the brackets may be dropped and we may write, e.g., AGE 1.

Regrouping can be effected by the use of the symbols +, —, V, and .. These are used as connectives to produce a new group from one or more previously defined groups. Thus we may form a new grouping, MIDAGE, say, by any one of the (alternative) statements:

MIDAGE = AGE 3 + AGE 4/AGE 1 +
AGE 2 + AGE 5 (1)

MIDAGE = AGE 3 V AGE 4/AGE 1 V
AGE 2 V AGE 5 (2)

MIDAGE = AGE (3 + 4)/MIDAGE—1 (3)

Any of these statements imply that persons with ages in the range 35–64 will be treated as belonging to MIDAGE 1, whereas all others will be considered to belong to MIDAGE 2.

Since no person can have more than one age, and must thus belong to one and one only of the groups under AGE, the + sign and the V (or) sign are interchangeable. The symbol + always has its natural meaning, and, if it is used on two groups A and B which are such that a single answerer can belong to both, the result of A + B will be to count that answerer once under each group. The expression A V B, however, will arrange to count an answerer once only whether he belongs to group A or group B or both. The — sign is used in the sense of negation, and MIDAGE — 1 is thus "not MIDAGE 1."

The symbol . (and) is used to define new groups, membership of which requires simultaneous membership of two other groups. Thus A.B refers to all those

answerers belonging to both A and B. We note that, if A and B are mutually exclusive, the group A.B is empty. It is not normally permitted to have more than one of the symbols +, V and . in the same statement.

5. Tabulation Statements

It is assumed here that tabulations are to be presented on a printed page. We may select the population with which the tabulation is concerned, the labelling of the rows, and the labelling of the columns. In order to present tables involving complicated subdivisions of the data, it is sometimes desirable to group the rows or columns and repeat a particular sequence within the elements of another group.

Thus we may wish to form a table as follows:

	Age Group 1		Age Group 2		Age Group 3	
	Men	Women	Men	Women	Men	Women
London						
Bradford–Leeds						
Manchester						
Glasgow						

To define such a table in Autostat language, we begin by stating in terms of labels the population from which informants are to be drawn, e.g. ALL, MEN, A.B, Q10, etc. Following this we state the row layout × the column layout. Groups tabulated within other groups are distinguished from one another by a solidus, /. For instance, assuming age groups to have been set up by a statement such as that defining AGE above, and the labels SEX and TOWN to be defined by

TOWN = Q15, LONDON/LEEDS V BRADFORD/MANCHESTER/GLASGOW

SEX = Q16, MALE/FEMALE,

the table given above might be defined as

ALL = TOWN × SEX/AGE.

More than two solidi on either side of × are not permitted (and would normally be impracticable as regards presentation). It is possible to conjoin two classes under the solidus and write

TOWN × SEX, SOCIAL GRADE/AGE.

Two tables may be conjoined in the same statement, if they have the same row or column layout, by use of the symbol +. Thus we may write

ALL = TOWN × SEX/AGE + WEIGHT

to denote the two separate tables of the form

TOWN × SEX/AGE and TOWN × WEIGHT,

where a set of weight groups have been defined elsewhere. The tables will be presented contiguously.

Direct regrouping of subgroups of a tabulation group may be effected within the statement. Thus we may write

ALL = SEX × AGE (3 V 4, 1 V 2 V 5)

to produce a table set out as follows:

	Age Group	
	35–64	Other Ages
Men		
Women		

However, the restriction that not more than one of +, V and . may appear in a statement applies here.

If the whole population is to be counted, the population designation (ALL =) may be omitted entirely. All tabulation statements must be numbered for reference purposes, since both the weighting scheme to be used and the output format must be stated, and this is most conveniently done in separate statements, linked to the tabulation statement by a cross-reference.

6. Weighting

The combination of tabulation statements with those necessary for regrouping provides the information essential to the machine for selection, from among the whole population of questionnaires, of those which, for a particular tabulation, are to be counted. We require also to specify the weight which is to be given, in counting, to each questionnaire or group of questionnaires. It is essential to be able to specify (1) the group to which a weight will apply and (2) the tabulation for which the weighted count is needed. Furthermore, the actual weight may be stated either as a number directly specified, as the answer to a question on the questionnaire, as a derivative of two or more answers, or as a number fixed by reference to an unweighted count of those questionnaires belonging, normally (but not always), to the group being weighted. Provision must thus be made for any of these methods of weighting or any combination of them. This is achieved in Autostat by a series of weighting statements.

An example of such a statement is:

1) A.B = 2·5

This would imply that all those questionnaires possessing both the properties A and B, and which are involved in tabulation 1, are to be weighted by a factor of $2\frac{1}{2}$. Omission of the 1) would apply the weighting scheme to all tables. More than one statement may be made about each tabulation. In this case both weighting schemes are applied, the product of the weights applying to the overlap of the two groups involved.

Thus, if we have also

1) C = 0·1,

questionnaires possessing property C will be weighted by 0·1 unless they also belong to A.B, in which case they will be weighted 0·25. Questionnaires in A.B but not in C will be weighted 2·5. All remaining questionnaires involved in tabulation 1 will be weighted 1·0.

In order to specify the use of the answer to a question

as a weight, the symbol n or V can be used following the label specifying the question, whether this be a Q-label or not. Thus we may write, for instance,

1) A = SALES V

or 1) A = Q7 V

If Q7 is "What is the value (in sterling) of the goods sold?" then all of the questionnaires in tabulation 1 possessing property A will be weighted by the actual value appearing as an answer to that question. This enables, for example, a sales total over a set of invoices to be obtained by the use of the system.

In all weighting statements, the left-hand side of the equality refers to the group to which the weight is to apply, and the right-hand side is numerical. It is possible to specify algebraic operations between numbers on the right-hand side, using the symbols $+$, $-$, $\times$, and /. The solidus here implies division of the number preceding it by that following. Not more than one operation of multiplication or division can be incorporated in a statement, but there is some freedom to use $+$ and $-$ in conjunction with this to add together answers to questions, and multiply or divide them by other answers.

Weighting schemes involving preliminary counts are to be dealt with by reference to the elements of the table which defines the preliminary counts, i.e. by reference to its tabulation statement. No provision is to be made for this in the first model of Autostat, but it is expected to be incorporated in later versions. For the present such weighting schemes are achieved by punching out the preliminary results on paper tape and then reinserting them to form a weighting table. Calling in such a table from paper tape is achieved by a statement of the form

1) A = TAPE

or 1) A = TAPEB

according to the reader being used, in a manner similar to Autocode practice (see, e.g., Felton and Clarke, 1959).

Apart from the possibility of several weighting schemes being superposed on a single table, it is sometimes desirable to carry out the same tabulation with different weights applied to the questionnaires involved. This is provided for in the system without the necessity to write out a second tabulation statement of identical form. We write, for instance,

1.1) D = 7·0
1.2) F = 2·7

to indicate that two separate counts with different weights are to be computed for Table 1. The two counts are carried out simultaneously by the computer, which also automatically finds the sums of the squares of the weights.

7. Formats and Statistics

Formats and statistical routines are to be specified by special statements comprising a name and a set of parameters. In the first version of Autostat these facilities will not be provided in the language, and routines will be written "ad hoc" for printing from or analysing the results on the magnetic tape put out from Stage 2.

It is not the intention to provide, at present, a language for the construction of such routines as required, but rather to regard them as a library of routines which can be drawn upon by virtue of a statement in Autostat language of the form, e.g.

1) FORMAT 2

or 1) ANALYSISOFVARIANCE (A.B/C).

By the first statement we understand that the format routine numbered 2 is to be used for Table 1, and by the second, that analysis of variance for groups A and B within C is to be carried out relating to Table 1.

8. Checking during Input

During Stage 1 we require to apply certain tests of validity to the data. One class of such tests has already been implicitly considered in the statements specifying the Q-labels in terms of input, i.e. "ring" tests. Thus if we write Q8, 16–24 we exclude answers not within the limits specified.

The statement defining AGE in a preceding paragraph would, for instance, be adequate to ensure the rejection of questionnaires on which ages below 16 or over 110 appeared. This form of check can be used to detect gross errors in the data.

A further type of check frequently useful is the "redundancy" or "internal consistency" check. If redundant information is supplied, so that the answering of a question, Q5 say, in one sense implies that the answer to another question, Q8 say, ought to be in a particular sense, then we may check for this during input. The application of such a check may be specified in Autostat by a statement such as

Q5 (2) → Q8 (1)

or SEX 1 ≠ MOTHER

The first statement implies that, if the answerer's reply to Q5 puts him in group 2, then his answer to question 8 ought to put him in group 1. If this is not the case, the computer makes a note and rejects the questionnaire pending further instructions; reading does not stop. The second statement implies that if the questionnaire belongs to the group SEX 1 (which might include all males, for instance), then it should *not* belong to any of the groups labelled MOTHER. Any discrepancy would be dealt with as described above.

9. An Example of the Use of the Language for Survey Work

Let us suppose that we are conducting a survey concerning the readership of newspapers. There will be, in essence, one similar question asked with respect to each newspaper in the questionnaire, namely whether the interviewee reads that particular paper or not.

Other questions would refer to the age, sex, area of domicile, and personal qualities of the interviewee, e.g. whether he is a smoker or not. Thus we construct a hypothetical questionnaire as in Fig. 1. Let us suppose we now require to provide a direct count of the number of smokers who read each of the newspapers covered by the survey, smokers being subdivided into those who smoke a pipe only, those who smoke cigarettes only, and those who smoke both, the whole table being further subdivided into separate sections relating to males and females, each sex being grouped by age groups 16–24, 25–34, 35–44, 45–64 and 65 and over.

In order to set out the tabulations asked for, we may write down:

```
TABLES
1) MEN = READERS × SMOKERS/AGE
2) WOMEN = READERS × SMOKERS/AGE
END
```

We now require to define the labels used above. We may do this as follows:

```
DEFINITIONS
Q3,  M/F
MEN      = Q3(1)
WOMEN    = Q3(2)
AGE      = Q2, 16–24/25–34/35–44/45–64/65–110
PIPES    = Q4.1,  YES/NO.
CIGS     = Q4.2,  YES/NO.
SMOKERS  = PIPES 1.CIGS 2/PIPES 2.CIGS 1/
           PIPES 1.CIGS 1
READERS  = TIMES 1/TELEGRAPH 1/GUARDIAN
           1/EXPRESS 1/HERALD 1/
           CHRONICLE 1
           TIMES      = Q1.1,  0/–
           TELEGRAPH  = Q1.2,  1/–
           GUARDIAN   = Q1.3,  2/–
           EXPRESS    = Q1.4,  3/–
           HERALD     = Q1.5,  4/–
           CHRONICLE  = Q1.6,  5/–
END
```

This would be adequate to produce an unweighted count. However, we require, perhaps, to weight all those liable for jury service by a factor of one half, to correct for the fact that twice as many have been interviewed as from among those not liable. We also require to correct for interviewing differences in various areas, for example.

In this case we must add suitable weighting statements such as

```
WEIGHTS
Q5(1) = 0·5
Q6(1) = 1·1
Q6(2) = 0·953
Q6(3) = 1·32
Q6(4) = 0·8
END
```

We must now also add some definitions of Q5 and Q6

Q1. Which of the following papers do you read, if any?
(Ring the numbers) Times 0
Telegraph 1
Guardian 2
Express 3
Herald 4
Chronicle 5
Q2. How old are you? (Insert figures in box.) ☐
Q3. Sex? (Ring) M. F.
Q4. (1) Do you smoke a pipe? YES. NO.
(2) Do you smoke cigarettes? YES. NO.
Q5. Are you liable for jury service?
YES. NO.
Q6. Area of domicile? (Insert figures, from key given, in box.)
☐☐☐

Fig. 1.—Hypothetical questionnaire

to the list previously prepared. This can be done by inserting them between END and the last of those definition statements before END, or by heading the new set, and writing,

```
ADD DEFINITIONS
Q5,  YES/NO
Q6,  0–127/128–143/144–150/151–200
END
```

A group of statements of this kind can be inserted anywhere in the sequence of statements, as can additional weighting statements, headed "ADD WEIGHTS" and closed by "END," or tabulation statements suitably headed and closed. Considerable flexibility of sequence is permitted since the whole Autostat program is read and decoded during Stage 0. Provided each statement is free from ambiguity, the sequence in which they are decoded is of no significance, unless paper-tape information is required in weighting, when a correspondence must be preserved between the sequence in the Autostat program and the appearance of relevant numbers on the tape.

If we wish also to include redundancy checks, we may do this by a further series of statements, such as

```
CHECKS
Q5(1) → Q2(n) ⩾ 21
Q4.1(1) ≠ WOMEN
END
```

These statements can, of course, be included anywhere in the sequence of statements. They will cause to be printed out during input the fact that they have failed, information from the questionnaire concerned being ignored. Information from the questionnaire, corrected if necessary, can be inserted subsequently if required.

Provision is made to stop and restart a program anywhere during its operation between dealing with consecutive questionnaires. On restarting it is not necessary to feed in the Autostat program again, since the relevant parameters are stored, after translation, on magnetic tape, and can be called in either by the use of a simple steering sequence, or manually. Care must be taken,

however, if a stop is made in the middle of the operation of reading in weighting information from paper tape.

10. Conclusion and Acknowledgement

Precise details of the language have been omitted in this paper in some places. These omissions have been made, not only to simplify the explanations given of the structure and usage of the language, but also because it is anticipated that later versions of the scheme on machines other than Pegasus will differ materially in some details, though not in outline, from the version being developed now. Several interesting points have arisen in producing a Pegasus program for providing Autostat facilities; these will be dealt with in a later paper. Use of the scheme will also be illustrated by reference both to a large-scale readership survey, and to costing and sales analyses. Work on both types of application, and on extension of the scheme to handle retail audit procedures, is currently in progress.

It is a pleasure for the authors to acknowledge their indebtedness to the British Market Research Bureau and, in particular, to Dr. J. A. P. Treasure and Mr. J. Fothergill of that firm for assisting in the initiation of this work and for encouraging its development.

Reference

CLARKE, B., and FELTON, G. E. (1959). "The Pegasus Autocode," *The Computer Journal*, Vol. 1, No. 4, p. 192.

Correspondence

The Editor,
The Computer Journal.

Sir,

May I comment on the paper by K. T. Boyd on "Simultaneous Equations and Linear Programming" in your April issue?

The use of the Simplex method for the solution of simultaneous equations and inversion of matrices was first suggested by Orden (1); practical applications were made on the Ferranti Mk. I computer at Manchester University in 1953 (2). Here are some details of the method used.

The machine starts by computing the row sums

$$\sum_j a_{ij} = R_i$$

of the given matrix (a_{ij}) and then reverses the sign of all the rows for which this sum is negative, keeping a record of the sign reversals.

This results in a modified matrix (a'_{ij}) with non-negative row-sums R'. For this matrix, the machine computes the column sums

$$\sum_i a'_{ij} = K_j$$

and then solves the linear programming problem:
Maximize

$$z = \sum_j K_j x_j \tag{1}$$

subject to

$$\sum_j a'_{ij} x_j \leqslant R'_i. \tag{2}$$

The set of variables

$$x_1 = x_2 = \ldots = 1 \tag{3}$$

satisfies relations (2) as equations; there are no slacks. That it constitutes an optimum solution can be seen as follows. Put

$$\sum_i R'_i = \sum_j K_j = \sum_{i,j} a'_{ij} = S.$$

Summation of (2) over i shows that

$$\sum_{i,j} a'_{ij} x_j = \sum_j K_j x_j \leqslant S$$

or

$$z \leqslant S$$

i.e. that S is an upper bound for z.

But this upper bound is reached when (3) is substituted in (1), proving that (3) is indeed the optimum solution.

In practice, the "contracted" version of the simplex method was used so that matrix (a'_{ij}) was replaced by its inverse, apart from certain permutations of rows and columns. The correct order of rows and columns was restored during printing out; at the same time, the sign of those columns, for which corresponding rows in the original matrix had undergone a change in sign, was reversed. The result was the required inverse of the given matrix.

The programme was used successfully for the inversion of matrices too badly conditioned to be inverted by other programmes available at the time, but it is not known whether this was due to some peculiarity of the simplex method or to the fact that an unusually large number of digits was employed in the computation.

Yours faithfully,

D. G. Prinz.

Ferranti Limited,
West Gorton,
Manchester, 12.

13 June 1960.

REFERENCES

1. ORDEN, ALEX. "Application of the Simplex Method to a Variety of Matrix Problems," *Symposium on Linear Inequalities and Programming*, Washington D.C., 14–16 June 1951.

2. PRINZ, D. G. "Some Experiences on the Manchester Computer with the Simplex Method," *Linear Programming Conference*, London, 4 May 1954.