Historical Software Issue 10: Statistical Analysis System (SAS)

Thaller, Manfred

Veröffentlichungsversion / Published Version
Zeitschriftenartikel / journal article

software

Issue 10: Statistical Analysis System / SAS

General Considerations

SAS is reviewed quite favorable by most people; indeed the only reason why it is not recommended more generally is, that is has that far been restricted to IBM systems - and IBM is definitely not very popular among academic computing centres(1).

SAS is a very large system: the documentation used for this review fills 12 volumes and even more information would be available from additional technical reports(2). Looking quickly through those 3000 pages of information, the first thing one notices is, that SAS is by no means restricted to statistics proper. There are e.g. system components for writing letters (SAS/FSP p. 37 - 59) or the production of mailing labels (Basics p. 623 - 628) and one can count upon the services of a whole family of rather sophisticated Full-Screen facilities (SAS/FSP) plus a plotting system (SAS/Graph), which includes already drivers for the more prominent plotting devices.

In the documentation SAS is referenced not so much as a statistical package, but as a programming language(3); in our opinion one should describe it better as a core of sophisticated statistical programs embedded in a consistent software environment. As a result of this design SAS programming falls more or less apart into two categories: on the one hand the statistical procedures can be used without any technical expertise; on the other hand everything that goes beyond that, particularly data handling, can relatively soon reach the limits of the canned solutions which are provided by appropriate options of the relevant commands. The limits of the canned solutions, not those of SAS, we should emphasize: the data manipulation statements of SAS are extremely powerful and when SAS claims them to form a higher programming language, there is few reason to dispute this claim. The problem of course is, that using such possibilities, one has to know a bit about programming: the solution SAS offers is a very powerful macro facility, which allows more sophisticated users to write within the command language of SAS programs which can be applied by their less computerwise colleagues as easily as the canned solutions provided by SAS. So this system fits perfectly into research setups, where a relatively small number of computational experts support a larger number of users which write their own programs, but do not have any in-depth knowledge what has to be done to prepare data for their analytical interests.

This policy of providing an environment(4), which solves the standard problems immediately and presents very powerful and convenient instruments to tackle more sophisticated cases, is taken even further:
SAS actively supports users who want to integrate their programs into the system. There is a separate volume of the system documentation devoted exclusively to the question "how to integrate your own programs into the SAS system", containing very detailed technical information (Programmers).

Address all communications to: Manfred Thaller, Max-Planck-Institut für Geschichte, Hermann-Föge-Weg 11, D-3400 Göttingen

All this does not mean, that SAS would not provide ready made programs: the system provides an ample set of powerful statistical routines. The greatest shortcoming of SAS from an academic point of view certainly is that it was restricted that far to IBM mainframes; currently SAS is changing this policy and a drive towards a "portable SAS" has been inaugurated. Right now SAS is spreading towards mini-computers. SAS itself considers versions for all 32-bit machines possible (verbal communication from SAS Germany). This reviewer has to admit, that looking through the examples of source codes and proposed programming techniques in the SAS Programmers Guide and having some experiences with the restrictions of PL/1 compilers at various machines he became doubtful if the spreading of SAS to other brands of computers is going to be particularly fast. Anyway, SAS is determined to become available on other machines and that in its full, impressive range: a release notice for portable SAS in the European Political Data Newsletter 48 (September 1983) p. 60 admits, that in the very first release a couple of features will be missing, but lists among the ones available very sophisticated ones already and claims, that the "full" system will be available in 1984 in a portable version.

a. Statistical Methods available

As an example for the general layout of the command language we can use the command to compute a correlation coefficient.

```
PROC CORR;
```

will compute (Pearson) correlation coefficients between all possible pairs of numeric variables in the "current input data set" (i.e. that file (of potentially many others) you referenced as the latest one during the current job you submitted to SAS).

```
PROC CORR DATA=PEOPLE;
  VAR INCOME AGE;
```

would compute the Pearson correlation between INCOME and AGE in the SAS data set (= systemfile) PEOPLE. The format of SAS commands is (with few exceptions) free, so

```
PROC CORR DATA=PEOPLE; VAR INCOME AGE;
```

would be equivalent to the command above.

```
PROC CORR SPEARMAN; VAR EDUCATION TAX_CLASS;
```

makes the same command useful for ordinal variables, computing Spearmans rank rather than Pearson correlations. A nice feature (available with virtually all statistical procedures) is furthermore the way to process subfiles:

```
PROC CORR; VAR INCOME AGE; BY SEX;
```

will compute separate correlation matrices for the two sexes, without the necessity to count their frequency first and put them into separate subfiles. The only shortcoming: you have to take care yourself, that the data are sorted according to your BY variables - as SAS went so far in improving subfile handling, it might have been possible to keep track of all sorting done on a file and presort the data set automatically when a BY comes up, that is not equivalent to the current order of the data set. But this comment must not obscure the very crucial fact, that the mentioned possibi-

lity to use BY with virtually all statistical procedures makes the analysis of certain types of subgroups and the controlling of third (fourth and nth) variables considerably easier indeed.

So much to give you a flavor of the language used for the activation of the statistical routines. A lot of descriptive statistics are available with many nice details: UNIVARIATE (computing the parameters of the distribution of a set of variables) provides one with a whole set of quantiles; the FREQ command very cleverly sees a one dimensional frequency distribution as a one dimensional table and provides at the same time for crosstabulations as well, giving you considerable control over the layout of your table and the values you put into the boxes.

Needless to say that SAS provides a broad range of tools for analysis of regression and of variance, including tools for the less well known types of it, as non-linear regression (Statistics: p. 15 - 37) or non parametric approaches in the analysis of variance (Statistics: p. 205 - 211), and that the well known multivariate techniques are supported (Canonical Correlation, Factor Analysis and Principal Component analysis) (Statistics: p. 293 - 361).

What is much more interesting for students of historical social research is, however, that a large number of methods are provided, which are not so well supported by other packages and very appropriate for analysis of historical sources: Cluster analysis is available (though with a very limited number of methods available when compared to more specialized software like CLUSTAN (Statistics: p. 415 - 473; compare S-124) and Discriminant analysis is supported very well (Statistics 363 - 414), comprising a couple of approaches this reviewer considered to be part of Cluster analysis that far. Most important: SAS provides two commands (Statistics: p. 245 - 292; compare Library: p. 83 - 102) to deal with categorical data, i.e. with data on the nominal scale, which are very powerful indeed, allowing for mixed analyses as well (i.e. treating multivariate relationships between nominally scaled variables and such on interval scales). And, at least for the more simple cases, these commands can be used with a command language, that is as easily to be understood as the one employed for more traditional approaches.

Time Series Analysis is supported very well, ranging from facilities for printing time series prettily right through to (simple) spectral analysis and facilities for simulation, which, when accessible that easily, could very well be applicable to historical studies dealing, e.g., with problems of transfer within a community (SAS/ETS).

One can summarize that SAS provides a spectrum of statistical methodology that contains quite a number of tools which would be specifically useful for the analysis of typical problems of historical social research and could markedly improve the quality of quite a few quantitative studies. Still, this reviewer thinks, that the undisputed statistical power of the system is not necessarily what merits most of the attention. Indeed some of the packages we are going to discuss in this series may rate better here, particularly BMDP. But the integrative approach of SAS has provided maybe its most interesting result: SAS provides a standard interface to BMDP, so, using SAS for manipulating your data, which is perhaps the most important strength of the package, you can submit your data to BMDP, the most encompassing system seen purely statistically, for analysis with any of its routines, without leaving SAS (5) (Basics: p. 665 - 671).

b. Learning and Teaching

The documentation of SAS is excellent. The Introductory Guide claims, that without previous exposure to data processing one should be able to handle the system within 10 minutes (Einführung: Vorwort). This probably somewhat overstates the case. But as the control language is very flexible,

not too vulnerable against formal errors, and because the simple varieties of the data description commands have rather obvious meanings, it should be possible to start producing results within a few hours.

In principle SAS fulfills the conditions for a good package for teaching we defined in the last issue (ready made routines, but a language powerful enough to show the beginner how to compute parts of such a result by explicit commands). In practical teaching, though, those parts of the command language, by which one could accomplish this, will be too complicated for most beginners.

So: a control language easily learned and easily to be used when one sees the computer as a tool for producing results; an insufficient one, if it shall be a tool for teaching statistical methodology and understanding.

c. Enhanced Data Handling Capabilities.

This may be the highlight of SAS. The basic logic runs like this: The system ditterentiates sharply between preparing data and using it. So, every statistical procedure ("or PROC step" for PROCEDURE, the command triggering all statistical analyses) deals with a data set, that has been prepared for this analysis and has been stored in a file before it starts.

This preparation and storage is done by a "DATA step". These steps can be arbitrarily mixed and repeated, referencing within one SAS program a (theo-retically) unlimited number of data files, merging them, splitting them into subfiles and running different analyses on the different files created, writing intermediate results as a by-product of an analysis to a file, preparing them with another DATA step for another PROC step, and so on.

Basically data steps can be written down as easily as PROC steps:

```
DATA;
 INPUT AGE 1-2 SEX 4
 CARDS;
      :
 Lines/Cards with input data
      :
PROC FREQ;
 TABLES AGE*SEX;
```

being all that is needed to define input data containing age values in columns 1 and 2 and a sex code in column 4 of successive lines/cards, producing a crosstabulation out of those two variables (and if certain JCL commands have been given beforehand, even saving the data set created).

One would fail, however, to do justice to SAS, if one would try to describe the data definition/data modification language of SAS as just a collection of commands like INPUT or CARDS. The important point is, that besides ready made commands like these two you can use in the data step, it is practically a higher programming language: you have at your disposal block oriented control structures, macros (which you can use as subroutines), can give explicit read and write commands and for particularly sophisticated applications even access to single bits.

For computing and recoding of variables many helps are prepared: SAS provides well beyond hundred functions starting with the usual ones like ROUND() (rounding) or MAX() (largest one out of a set of numbers), continuing with an impressive number of mathematical, probability and random number functions, including a full set of character manipulation functions and providing finally a large number of special purpose ones, of which not less than 22 are dedicated to the treatment of calendar dates and time.

While you have no way to get the parameter of the distribution of a variable into a computation directly by entering e.g.

STANDARD=ABS(ORIGINAL_VALUE-MEAN(ORIGINAL_VALUE)),

there is a standard way how to use one of the procedures to reach (almost) that effect (Basics p.255).

Missing values (up to 27 different ones of them) are handled very consistently: if there is any variable in a computational expression, that has for a given observation a missing value, the value returned by that computational expression is missing as well. (Or expressed in terms of troubles of current SPSS users: if you enter the equivalent of a COMPUTE statement, the variable to the left of the equals sign is always missing, if any of the ones to the right are missing - no ASSIGN MISSING or IF/MISSING VALUES necessary).

SAS has been criticized not to provide overly comfortable means for the automatic detection of errors in input data: this is true, but there is a number of utilities, detecting e.g. lost or misplaced cards, and there are very sophisticated input formats checking implicitly for, e.g., the formal correctness of calendar data). And, if you want to consider the hardware dependent full screen facilities as part of the system proper, SAS even supports dynamic checking of input data by predefined masks on video terminals (SAS/FSP: p. 29 - 34). Still: in most cases you have to work with explicit IF ... THEN ... ELSE constructions.

Labelling features for variables and values are provided - without being extraordinary - they should be sufficient for most cases.

A highlight (though useful only for statistically rather sophisticated users and probably beyond the needs of most historical social researchers) of SAS is its MATRIX procedure: it provides the full matrix algebra in a command language that is a consistent extension of the one used for the DATA step, including a number of additional functions for matrix handling (e.g. Eigenvalues). Very many methods - particularly for the analysis of data on the nominal level - could be implemented easily with the help of this feature.

d. More Complex File Handling

SAS has excellent facilities for the improved handling of data, but it clinges still to the rectangular data matrix. There are no features for the implicit handling of the problems related with files containing hierarchies and/or networks of data. Still, SAS has been used by a number of people to analyze and administer very complex files.

This contradictory situation can be explained by our initial description: "a statistical system in a very general programming environment".

Four solutions for handling non-rectangular files exist:

- the data modification language of SAS is so powerful that approaches as the one presented by this reviewer in issue 2 of this section (HSR 20/September 1981) can be implemented relatively easily. Probably one could even define a macro library for data steps, which would turn SAS into a (not exactly extremely efficient) hierarchical data base management system.
- the wider software environment of SAS contains a very close link to an IBM specific data handling language (SAS/IMS-DL/1).
- for a few (commercial) data bases which are very wide spread in the USA there are explicit interfaces built into SAS.
- one can always realize an interface in PL/1 or FORTRAN and has than to program only the data management part of an application, still being able to use the full analytical power of SAS. As a result of this, a couple of special problems (e.g. how to read index-sequential files with a given set of keys) have already been solved by users and can be called upon from within SAS, provided the appropriate library is installed at a given

computing centre (e.g. Library 71 - 74; compare Basics 63).
Still: there are no ready made procedures for non-rectangular files. This reviewer considers this a serious shortcome.

Merging of data sets is supported quite well. For data sets as used in historical social research, one would like to have a slightly more flexible handling of situations when two observations in two files contain two variables with the same name, but different values: in such cases only the value from the "later" file is kept - an introduction of multiple entries into one variable would be a better solution. But: to do so, you would need a system, where a variable in a given observation is not a scalar but (potentially) a vector - and as large statistical systems go, this concept seems that far to be realized only within OSIRIS. But, as with the need to sort explicitly: this comment must not obscure the great power of the merging routines as they are.

e. String Handling in Statistical Packages

This heading is actually an understatement of what SAS can do in this area. The data modification language of SAS contains full handling of (unfortunately basically fixed length) string variables up to 200 characters in length and the system documentation contains examples how to handle files consisting only of texts of variable length(6). One could probably handle simple content analytical tasks within SAS.

Report Generation is supported. Besides a few rather specialized business applications, however, all but the most simple reports have to be created with the help of the data modification commands within a data step, that is, on the level of a higher programming language. While the SAS manuals contain many examples with very tricky reports (e.g. Applications p. 106: a report that prints in six columns of a lineprinter page information about age, size and weight of high school kids, printing for each of them a sketch of a human figure, visualizing the relative size and weight of the child) you have to have a lot of experience to use those facilities.

f. Improved Display Facilities

As we mentioned already, SAS contains its own plotting package, that can be used by an extension of the more general command language. This package is unusually well supported: drivers for quite a few important devices are included and the installation has additionally access to a rather large number of parameters, that should allow the installation of this plotting package in quite a few cases without any changes in the source code.

The language by which the plotting routines are controlled is fairly simple:

```
PROC G3D;
  PLOT AGE*INCOME=YEARS_IN_SCHOOL;
```

would be sufficient to produce a three dimensional plot of the relationship between these variables, plotting the surface they define (that is, the actual points are combined by lines) taking automatically care of all problems arising out of the overlaying of several parts of the curved area when it is mapped onto the two dimensional paper and even automatically indicating by color if you look at a given part of the plotted surface from "above" or "below" (SAS/GRAPH 97 - 98).

What - in accordance with our principles of review, laid down in the last issue - we consider even more fascinating, however, is that SAS contains a large number of routines, which draw approximations of the usual plot representations (pie chart, bar chart, star chart etc.) on a lineprinter. For very many of the figures you can plot on a suitable device there

is an equivalent routine that will print a similar sketch on a printer: even without taking into consideration, that many university computing centres have still insufficient plotting devices, this means that you can in any case produce extremely cheap previews of your plots and experiment a lot more than if you would have to restrict yourself to plotters alone.

EDA is basically unsupported. There exists a PROC in the user library, however, which draws Tukeys "Box-and-Whisker Plots" on a lineprinter. (Library p. 173 - 175.)

Conclusions

SAS is a remarkably powerful system. It has a command language easy to understand to begin with, but able to be applied to problems of almost unlimited complexity. If a scholar of historical social research has access to it, he has a very large number of possibilities beyond what has settled down more or less as the "SPSS canon of social-historical methodology". There are quite a few projects where the need for own software development might have been reduced considerably, if SAS would have been available on the local mainframe.

Still, SAS will be used most advantageously in a setup, where a community of researchers can draw upon the skill of one or more programmers tailoring special solutions (particularly for filehandling) out of the very general but low-level tools provided by the system. The researcher working alone will still have to look very carefully, if his or her experience is sufficient before he or she starts with a project that can in principle be handled by SAS, but may take a rather long time to fully realize for a person not familiar with higher programming languages.

Notes

1) SAS scores best or second to best on almost all features in the comparative tables in Ivor Francis: Statistical Software - A Comparative Review, N.Y. etc.: North Holland, 1981, p. 6, 7, 21 and 509 sqq. We would like however, not to overemphasize the importance of this very popular book, as we share (some of) the doubts expressed by Helmut Wilke: The Forgotten "User" in Statistical Package Evaluation, in: Statistical Software Newsletter 6 (1980) 2, p. 64 - 70. Rather more significant we consider the generally positive evaluations of most of the constituents of the package, even when compared to more specialized software. See, e.g., the place assigned to SAS in the following contributions to the second ZUMA conference on the scientific application of statistical software (Published as Helmut Wilke et al. (Edd.): Statistik-Software in der Sozialforschung, Berlin: QUORUM, 1983): Manfred Küchler: Kontingenztafelanalyse - Ein Software-Vergleich unter besonderer Berücksichtigung sozialwissenschaftlicher Programmsysteme, ibid. 25 - 70; Helmut Wilke: Datenmanagement mit statistischen Programmpaketen - Möglichkeiten und Grenzen, ibid. 211 - 258.

2) This review is based upon the following manuals, quoted in the text by the underlined abbreviations:
Jane T. Helwig: Eine Einführung in das SAS, Cary, NC: SAS Institute Inc., 1981.
Alice Allen Ray (Ed.): SAS Users Guide: Basics, Cary, NC: SAS Institute Inc., 1982.
Alice Allen Ray (Ed.): SAS Users Guide: Statistics, Cary, NC: SAS Institute Inc., 1982.
Kathryn A. Council (Ed.): SAS Applications Guide, Cary, NC: SAS Institute Inc., 1980.
Alice Allen Ray (Ed.): SAS/ETS Users Guide, Cary, NC: SAS Institute Inc., 1982.
Kathryn A. Council and Jane T. Helwig (Edd.): SAS/GRAPH Users Guide,

Cary, NC: SAS Institute Inc., 1981.
Patti S. Reinhardt (Ed.): SAS Supplemental Library Users Guide, Cary, NC: SAS Institute Inc., 1980.
SAS Technical Reports S-120 thru S-129, Cary, NC: SAS Institute, Inc., 1981.
Patti S. Reinhardt (Ed.): SAS Programmers Guide, Cary, NC: SAS Institute, Inc., 1981.
Virginia B. Sall (Ed.): SAS CMS Companion, Cary, NC: SAS Institute, Inc., 1981.
Allan P. Stone et al. (Edd.): SAS/IMS-DL/I Users Guide Version 1 Release 2, Cary, NC: SAS Institute, Inc., 1981.
Kathryn A. Council et al. (Edd.): SAS/FSP Users Guide, Cary, NC: SAS Institute, Inc., 1982.

3) Basics p. 76: "SAS does not treat trailing blanks as zeros as do some other languages (for example, FORTRAN)."
4) SAS goes rather far to accommodate users within a given operating system (See CMS). With all operating systems (of IBM) SAS can be used with, the system provides a whole array of utilities to take care of more general services, as e.g. an own editor (Basics: p. 715 - 733).
5) Equally impressing: SAS is able to process directly the system files of BMDP, DATA-TEXT, OSIRIS, and SPSS.
6) Basics: p. 203. Among the highlights of the documentation is certainly the Applications Guide which for all such "odd" case may be a source of inspiration how to put a command to some of its less obvious uses.