BASIC-PLUS programs for Sackett's lag sequential analysis

RICHARD DENI

*John F. Kennedy Center for Research
on Education and Human Development
Peabody College, Nashville, Tennessee 37203*

Sackett's (Note 1) nonparametric lag sequential analysis is one of a number of procedures to summarize and reduce observational data. Bakeman (in press) describes the lag sequential method and cites the data types common to observational techniques. His Type I data represents a sequence of mutually exclusive events. Two subsets of Type I data are possible: (1) *discrete events* where behaviors can follow themselves and (2) *event states* where behavior states cannot follow themselves. Event states assume that behavior categories comprise an exhaustive set of all possible events in a given situation (Sackett, Note 1, p. 1). Bakeman's Type II datum is an event sequence, as is Type I, but it includes concurrent behaviors. With Type III data, the duration of sequential events and the time block in which they occur are recorded, with the restriction that the events are mutually exclusive. In this category, behavior is recorded during discrete blocks of time. Type IV data add the option of concurrent events to the format of Type III data (see Bakeman, Note 2). With respect to the lag concept, Type I and II data establish lags with succeeding events; Type III and Type IV data establish lags with succeeding blocks of time.

Sackett's lag sequential analysis works best with Bakeman's Type I and III data. The programs described here are designed to perform analyses on these two types of data.

Pre-input. LAG0 is a program that allows the user to specify the numeric behavior codes (with their functional descriptions) that will be used in subsequent lag sequential analyses. This information is stored in a virtual core file (Digital Equipment Corporation, 1975).

Input. LAG1 is a program that processes the raw stream of sequential events. That stream can be an output file generated from a Datamyte (Electrogeneral Corporation) record or an entry from the user's keyboard. LAG1 creates a transitional output file which will hold the data for the lag sequential analysis. Three additional options are available when running LAG1: The user specifies (1) discrete event, (2) event state, or (3) time sequence analysis. When the user is entering behavior codes or codes and times from the keyboard,

LAG1 allows him to review recent entries and correct mistakes.

LAG2 is a program to allow the user to link together two or more transitional output files created by LAG1. This option is taken if the investigator wants to have repeated observation sessions combined for a single lag sequential analysis. The user enters the file names to be linked, and LAG2 chains them into a single file. Lagging does not continue past the end of each constituent file. The LAG3 program computes the conditional frequency of each behavior occurring at each sequential lag from a given criterion behavior. The user specifies: (1) either a transitional output file or a linked transitional file, (2) whether the analysis is for discrete event, event state, or time sequential data, (3) specific criterion behaviors, and (4) the number of lags. If the analysis is for time sequential data, the user must specify the duration of the standard time block.

Output. LAG4 is a program that computes the following for each behavior at each lag from the given criterion behavior: the expected frequency, the expected conditional probability, the observed conditional probability, and a chi-square value. This information is placed in a print file for later listing on a line printer. A table is printed for each criterion behavior that is put through the analysis. These tables are very similar to those used by Sackett (Note 1). A permanent record of the information is stored in virtual core files set up for each criterion behavior. The LAG5 program prints into a print file a series of graphs similar to those used by Sackett. They show a simultaneous plot of the expected probability and the observed conditional probability for each behavior with reference to the criterion across multiple lags.

Restrictions. Transitional output files and linked transitional files used by LAG1, LAG2, and LAG3 can be extremely large, holding a very long list of sequential event codes. They are limited only by the capacity of peripheral storage units (disk cartridge, mag tape, etc.). The virtual core files used by LAG3, LAG4, and LAG5 are limited by the size of core storage. The programs are written to handle up to 20 behaviors, for up to 10 lags. The major advantage of these programs is that the data set from which the analysis is performed is unlimited in size, since raw sequential codes are not placed in core.

Language and Computer. The programs are written in BASIC-PLUS and have been used on a PDP-11/40 computer under the RSTS/E operating system.

Availability. Program listings or punched tape versions can be obtained without charge from Richard Deni, Box 154, Peabody College, Nashville, Tennessee 37203.

REFERENCE NOTES

1. Sackett, G. P. *A nonparametric lag sequential analysis for studying dependency among responses in observational scoring systems*. Unpublished manuscript, University of Washington, 1974.
2. Bakeman, R. *Data analyzing procedures* (Technical Report 2). Atlanta: Georgia State University Infancy Laboratory, January 1975.

REFERENCES

- BAKEMAN, R. Untangling streams of behavior: Sequential analysis of observation data. In G. P. Sackett & H. C. Haywood (Eds.), *Observing behavior*. Baltimore, Md: University Park Press, in press.
- DIGITAL EQUIPMENT CORPORATION. *BASIC-PLUS language manual*. Maynard, Mass: July 1975.

(Accepted for publication February 22, 1977.)