Discussion

George Sadowsky, The Brookings Institution

I would like to comment on these three papers as a user of computers and as a tool builder rather than as a statistician.

The central concern of the papers is with the problem of identifying and/or constructing homogeneous population subgroups. Armor proposes a tool for computing the analysis of variance for models of a "bread and butter" type; a successful tool of this kind would lower the cost to researchers for doing these calculations. Sonquist is concerned with explaining variance by dividing the population into subgroups -- with the implication of later forming an explanatory model so that the subgroups formed by the procedure become increasingly homogenous as the procedure is extended. Hall is concerned more with simply recognizing clusters that exist, regardless of the homogeneity of the characteristics of observations within a cluster.

Armor's paper contains an error of modesty that is worth correcting. In his introduction he states that it was the demand for high-level languages for social science computing that has caused them to evolve. My observations indicate that this is not at all the case; rather, the supply of such languages has led the demand for them, and it is to people like Armor that we owe our thanks for producing these useful tools -- especially considering the lack of academic recognition given to this activity.

Armor's paper appears to be a nice integration of a widely-used class of analysis of variance design into the existing Data-Text language. There are some syntax alternatives that I would prefer to those in his paper, but the differences are relatively minor and I agree with the spirit of his construction. A larger issue is the condition under which Data-Text type language development is itself worthwhile. The principal alternatives to such development are (1) the existence of general programming languages such as Fortran, (2) languages of high-level operators such as APL and the Lincoln Labs RECKONER, and (3) interrogative systems rather than declarative languages.

The case for the development of specialized computing tools appears to be a strong one. Current price trends within the computing industry indicate that the roles and costs of hardware and software are rapidly reversing. In this sense Data-Text can be regarded as an investment in capital which once created, has a low marginal cost of distribution (unlike hardware) and high marginal benefit. The case for declarative languages is weaker, but it is certainly true that Data-Text type tools operating in batch environments are currently very useful and will be for some time. Furthermore, the competition between the very different approaches of interactive systems and declarative languages is likely to be strong and will yield benefits to social science computing as a whole.

Other relevant points of Armor's paper are: (1) missing data is handled; having to deal with missing data is not at all aesthetically pleasing for the systems designer, but crucial to widespread and general use of the system; (2) the problem of an "escape" into more powerful languages to handle extensions is not visible (such as extensions to a Latin square design), and although manufacturing "escapes" is difficult, it is useful to try to provide open-endedness; (3) intuitive language structure such as is employed by Data-Text may be easy to use, but consideration should also be given in formulating syntax rigorously, both for definition and for potential implementation using meta-compilers.

Sonquist's paper explores several additions to his already well-known AID algorithm. I have used AID, and would place substantial value on deepening the search strategy. Although I have not -- at least knowingly -- had occasion to be concerned with a covariate in our analysis, I would think that it might be a quite useful addition.

Sonquist's paper fascinates me most because of the possibility of extending the AID algorithm even further within an interactive environment. Admitting that the detection of interaction effects is important in data exploration activities, why is it that such a process must be automatic? The primary reason for the automation of the process is that most computer centers -- both formerly and now -- operate primarily in the batch mode. Some of the implications of this environment for interaction detection are less than satisfactory for social scientists. For example, if two variables have nearly the same explanatory power for a given split, the one with the greatest explanatory power will always be chosen even though the analyst may have good reason to choose one of them a priori from a knowledge of the data. In the same spirit, having to set a filter parameter value to distinguish "signal" from "noise" is difficult a priori when one does not know the "signal to noise" ratio of his data. Observing the branching process interactively would allow the analyst to terminate his

tree structure along various branches when it became apparent that no substantive explanatory gain could be obtained by going further. Furthermore, while the concept of expanding both the capacity of each node and the search strategy of the algorithm is laudatory, the cost increases exponentially. With some human guidance and pruning of the tree, the increase in cost of such improvement might only be moderate, and the benefits would be substantial.

I would suggest as an alternative a "guided interaction detection" algorithm embedded in an interactive environment. Such an algorithm could have several modes, such as automatic, semi-automatic, and manual. The automatic mode has already been implemented. The semi-automatic mode would be identical to the automatic mode except when the explanatory power of the best variable for any split was not decisively greater than all other explanatory variables and when the best reduction in variance obtained for any split decreased below a certain level. In the manual mode, the program would display the partitioning choices and the corresponding variance statistics for the analyst at each node and allow him to select the partition or form new candidate variables. The manual mode of operation would also allow "backtracking" upon the discovery of any evidence suggesting that a previous split might have been less than optimal, and it would also allow transition to any other procedure at any node or leaf of the tree -- such as covariance analysis, multiple regression analysis, or multiple classification analysis. Transfer between modes of operation could be simple and could be effected at the analyst's discretion. A "guided interaction detection" program would seem to be an extremely powerful tool for data analysis in the social sciences.

My first reaction to Hall's paper was to marvel at its convenience and elegance. I think it would be a very stimulating afternoon for me if I were in Palo Alto and had in my possession a set of data for analysis. Hall's approach seems to be more agnostic than AID, since it imposes no a priori structure on the data. Its main purpose appears to be to obtain cohesive or compact groups.

However, once PROMENADE or ISODATA has been used by the researcher, he may feel that only part of his analysis has been performed. While it is true that application of either PROMENADE or ISODATA causes subgroups to be formed and group profiles and other summary measures to be constructed, how then can the analyst move toward a model of the world he observes? Each group must be characterized, i.e., one might feel compelled to build models to explain group membership. Furthermore, if clustering is not of a very definite pattern -- either visually or actually -- the idiosyncracies of the data may cause group membership to be misleading. At the present time, the cost of a dedicated CDC 3100 computer with display is relatively high for this type of application. Although such costs will certainly decrease, they will not decrease as fast as those for simpler input-output devices that would support a guided interaction program. On the other hand, the experience gained by the use of any interactive system has a subjective quality which is hard for a reviewer to measure without having actually used the system. In the case of PROMENADE, the subjective impact becomes even harder to measure due to the extensive and elegant graphic capabilities of the system.

After reading both Sonquist's and Hall's papers, I found myself as a user wishing for some sort of a middle ground between them. They differ in that Sonquist's algorithm maintains an increasingly homogeneous characterization of subgroups formed, whereas Hall and his colleagues are more interested in data classification using metric criteria rather than in the characteristics of the subgroups. Data populations fall into a spectrum with regard to homogeniety. At the one end, there is highly clustered data, and at the other end, very diffuse data. I would prefer to use Hall's method on the former and Sonquist's on the latter.

Hall's interactive environment could be extended to allow the homogenizing of clusters based upon more analysis. For example, a researcher might prefer to use PROMENADE to isolate clusters of observations and then interactively analyze the characteristics of the clusters. He might also like to ask questions such as what the increase in unexplained variance might be if observations having certain characteristics in some clusters would be moved to another cluster. The answer to this question might determine whether he would choose to have the system move those observations and then make itself available for more such analysis.

The Monte Carlo experiments cited by Sonquist seem very worthwhile in evaluating the utility and power of these methods. One could conceive of a series of such experiments on data having different characteristics of diffusion -- perhaps using the same sample population. This would help to determine the sensitivity of these algorithms and their various modifications to "noise" in the data and it would help to determine the useful ranges of application of these systems.