Evaluation of a Semi-Autonomous Assistant for Exploratory Data Analysis

Robert St. Amant
Department of Computer Science
North Carolina State University
Box 8206
Raleigh, NC 27695-8206
stamant@csc.ncsu.edu

Paul R. Cohen
Computer Science Department, LGRC
University of Massachusetts
Box 34610
Amherst, MA 01003-4610
cohen@cs.umass.edu

ABSTRACT

AIDE is a knowledge-based planning assistant for intelligent data exploration that draws on research in mixed-initiative planning and collaborative systems. AIDE incrementally explores a dataset, guided by user directives and its own evaluation of the data. The system is mixed-initiative: it semi-autonomously pursues high- and low-level goals but allows the user to review and potentially override its decisions. This paper briefly describes the exploratory task, AIDE's architecture, and how the system interacts with the user. The bulk of the paper is devoted to an experiment in which we compared the performance of human subjects analyzing data with and without AIDE. Although subjects each worked with AIDE for only a few hours, the system clearly influenced the efficiency and coherence of their exploration. We surmise that AIDE facilitates data analysis primarily by helping analysts navigate through the large space of decisions involved in exploring a dataset.

Category: Paper.
Application area: Expert assistants.
Technical issues: Action selection and planning; collaboration between people and agents.

Introduction

Exploratory data analysis, or EDA (Tukey 1977), has come to play an increasingly important role in statistical analysis. Modern computer-based statistics packages contain a rich set of operations, suitable for almost any EDA application. One can transform and reduce data, identify and describe clusters, fit lines and higher order functions to relationships, build a variety of models to fit data, among many other possibilities. Results of these operations combine to provide a more complete picture of patterns in a dataset.

Unfortunately, EDA can be difficult. Conventional statistics packages offer hundreds of operations, which must often be combined in lengthy sequences to produce useful results. Choosing appropriate operations can depend on what one knows about the data—whether a linear relationship between x and y is plausible, if a natural interpretation of clusters exists, whether a set of values has a practical bound. EDA is hard for a human analyst to do alone, but because it requires informed, human judgment, complete automation of the process is not feasible either. To solve this problem a system must strike a balance between *autonomy* and *accommodation*, where "accommodation" means a responsiveness to knowledgeable human guidance (Lubinsky & Pregibon 1988).

For the past several years we have been working on a system to help people with EDA. AIDE, an Assistant for Intelligent Data Exploration, is a knowledge-based planning system that incrementally explores a dataset, guided by user directives and its own evaluation of indications in the data. As a mixed-initiative planning system (Allen 1994), AIDE must *assist* in an exploration, rather than taking over the process completely or waiting for instructions for each of its actions. AIDE balances autonomy and accommodation within a partial hierarchical planning framework, which generates an explicit representation of the exploration process for the user's review and modification. Our experience with this arrangement has shown it to be a promising approach to improving data analysis.

Elsewhere we have described AIDE's operations and primitive data structures (St. Amant & Cohen 1996a), its planning representation (St. Amant & Cohen 1996b), its user interface (St. Amant & Cohen 1997), and the system as a whole (St. Amant 1996). This paper discusses a recent evaluation we conducted with the goal of demonstrating that AIDE works. A closely related concern was that we understand *why* AIDE works. That is, AIDE is a large software artifact, under development since 1993. It has grown far too complex for us to say, "AIDE is effective for such-and-such a reason; proof by inspection of the code." Instead, we have built empirical models of the system's behavior and its interaction with users. This paper focuses on the development of these models and how they help us to better understand AIDE as an automated assistant.

Data exploration and AIDE

AIDE is designed around a partial hierarchical planner (Georgeff & Lansky 1986; Carver & Lesser 1993). Partial hierarchical planners are a type of reactive planner, designed for complex, rapidly changing environments. In many ways the EDA search space is such an environment: we can't plan every conceivable action in advance; each new nugget of information can change our view of the problem; a course of action we initially thought appropriate may suddenly become useless. Partial hierarchical planners have properties designed to help them cope with such environments. They can exhibit complex behavior, relying on precompiled, often hand-constructed plans to guide their actions. They are responsive to changes in the environment; they generate plans on the fly, rather than elaborating a plan to completion before executing it. Plan structures can be modified opportunistically in response to new information. This kind of planning is well-suited to exploration.

AIDE plans as follows. When a dataset or relationship is presented to the system, a goal is established for its exploration. The planner searches through its library for an appropriate plan and expands it, that is, establishes a set of new subgoals to be satisfied. These subgoals are satisfied in turn by plans from the library. Goals can also be satisfied directly by primitive actions, which execute code directly rather than establishing new subgoals. This process is more complex than it might initially appear: often, several plans in the library can satisfy a single goal, and there may be an unlimited number of ways to bind a given plan's internal variables to different values. For each decision, or focus point, the planner relies on a set of control rules to decide which plan or variable binding to select. As planning continues, the planner may sometimes backtrack to one of these focus points to make a different selection. The process continues until the goal at the top level has been satisfied. Thus we cast exploration as a problem of constructing and navigating through a network of decisions, represented by focus points. Execution of each primitive action generates one or more new results; the network of decisions and results combines all the findings.

This design provides an adequate framework for a (hypothetical) system that acts completely autonomously. At any point in the exploration AIDE considers one decision—a specific node in the focus point network—to be its current focus of attention. The system can make autonomous decisions based on its internal plans and control rules, without consulting the user. New findings are combined in the network with existing results, creating opportunities for further exploration. As control rules indicate that specific results should be pursued no further, then each branch of the exploration comes to an end.

As we argued earlier, however, complete autonomy is undesirable for AIDE. Rather, the system must accommodate the user's knowledge about the data and the goals of the analysis. Maintaining the balance between autonomy and accommodation is harder than might appear at first glance. If the system can take actions without consulting the user, it can as a natural consequence drive the analysis into inappropriate areas, possibly losing the user in the process. Similarly, the user must not be constrained to take only those actions the system finds reasonable; on the contrary, with better visual pattern-detection abilities and knowledge of what variable values actually mean, the user should be able to take the analysis in whichever direction appears appropriate. The system must nevertheless be able to follow in the user's path, ready to contribute to the process once it again reaches a familiar area.

AIDE's design addresses these concerns as well. Plans attempt to implement common statistical practice, relying implicitly on the user's knowledge of what actions are appropriate in a given situation. Thus AIDE's autonomous actions often anticipate the user's goals. (User modeling, which would allow reasoning about the knowledge and goals of *specific* users, is beyond AIDE's abilities.) More significantly, the user can explicitly direct AIDE's activities, the interaction taking place within a conventional menu-based statistics package. Menu choices let the user load a dataset, compose variables into relationships, compute summary statistics, generate linear models, partition data, run statistical tests, and so forth. These menu operations are tied internally to the focus point network, so that if the user tells AIDE to run a regression of y on x, the system will search through the network to find a decision point associated with selecting relationships, then find or create an appropriate branch for (x, y), and then incrementally select a sequence of plans that run the regression, explore the residuals, evaluate the results, and so forth. The user can review both the results and an explicit model of the decision-making process that led to the results, and make modifications as necessary.

Interaction is thus a flexible trade-off between autonomy and accommodation. When the user gives AIDE explicit directions to perform some operation, the system can follow along, ready to give assistance once that result is reached. When the user simply tells AIDE to carry out one of its own suggestions, the system proceeds autonomously, until it reaches a result it considers significant. At this point the user can review, extend, or redirect the analysis. The system behaves in some ways like a conventional statistics package and in some ways like an autonomous machine learning program, and in the best case combines the benefits of both types of system.

Evaluation

Evaluation focused on a simple hypothesis:

Exploration is more effective
with AIDE than without.

The experiment involved testing subjects under two conditions. In the USER+AIDE condition, subjects explored a dataset with AIDE's help, while in the USER-ALONE condition, subjects explored a dataset in a similar statistical computing environment, but without active interaction with AIDE. AIDE's effectiveness was then determined by measuring differences in performance between the two conditions.

Several factors can potentially confound an experiment like this: different subjects may have different facility with EDA techniques; user interaction may be different under the two conditions; the datasets to be explored may contain different types of structure and patterns; and the order in which conditions are presented to subjects may make a difference.

To control for these and other effects, we set up the experiment as follows. All subjects explored the same two datasets, one in the USER+AIDE condition and the other in the USERALONE condition. The interface was identical in both cases, lacking only AIDE functionality in the USERALONE condition. The datasets contained artificial data, generated by similar but not identical models, to provide equivalent problems to be solved in both conditions. The dataset/condition assignment was randomized, as was the order in which the datasets were explored. Because of the time and effort involved in overseeing individual trials, which lasted on the order of three to four hours per subject, the experiment was limited to eight subjects. The small sample size limited the range of the conclusions we had hoped to reach, but it did not affect our main results.

The generation of a dataset followed roughly this procedure: Start with a directed acyclic graph of twenty nodes. Each node corresponds to a variable. Associate with each node a simple function of the arcs from its incoming variables; for example, if a node c has arcs from a and b, the function might be $c = (a \times b - b) + \epsilon$, where ϵ is normally-distributed noise. Associate specific distributions with nodes that have no incoming arcs (exogenous variables). Compute a single row of the dataset by sampling from each exogenous node's distribution and "pushing" these values through the rest of the graph. By repeating this process many times, we can collect as many variates (i.e., rows of a data table) as we need. The two datasets for the experiment were generated from graphs almost identical in structure and with comparable distributions and functions attached to the nodes and arcs. Each variable in the dataset was given either a plausible-sounding name, to help a subject identify potential relationships, or an artificially generated name (e.g., "V20"). In the experiment, subjects were instructed to identify the direct relationships in the data and to supply observations describing them (i.e., as linear relationships, clusters, power relationships, and so forth.)

We defined several related measures of accuracy: the average number of direct relationships correctly identified and described, over all subject observations made ($\overline{p}$); the total number of correct observations ($k\overline{p}$); the average number of direct relationships correctly identified, without regard to their correct description ($\overline{i}$); the total number of correct identifications ($k\overline{i}$). Subjects performed as shown in Table 1. A matched-pair, one-tailed t-test tells us that $\overline{p}$ and $k\overline{p}$ are higher for subjects in the USER+AIDE condition: $t = 2.217$ and 1.808, with p-values around 0.03 and 0.06, respectively. A similar result holds true for $\overline{i}$ and $k\overline{i}$.

This comparison tells us that AIDE contributes significantly to the correctness of a given user's observations, on average, and that AIDE contributes to a higher total number of correct observations as well. To put this in perspective, we can dismiss a few plausible but trivial explanations for better performance in the USER+AIDE condition. First, subjects entered roughly the same number of observations in both conditions, with a median difference of 0.5 between the conditions. Improved performance thus depends not only on making more correct observations, but also on making fewer incorrect observations. Further, subjects directly examined about the same number of variables and relationships in both conditions: 73 for USER+AIDE, 66 for USERALONE on average per subject. This difference is not great enough to account for

	$\overline{p}$		$k\overline{p}$		$\overline{i}$		$k\overline{i}$	
	Aide	Alone	Aide	Alone	Aide	Alone	Aide	Alone
Subject 1	0.29	0.34	4.0	5.5	0.538	0.455	7	5
Subject 2	0.39	0.29	3.5	3.5	0.667	0.417	6	5
Subject 3	0.50	0.21	3.0	1.5	0.875	0.285	7	2
Subject 4	0.56	0.37	10.0	7.0	0.632	0.579	12	11
Subject 5	0.44	0.29	4.0	2.0	0.556	0.500	5	3
Subject 6	0.34	0.50	4.5	5.5	0.571	0.583	8	7
Subject 7	0.50	0.07	3.0	1.0	0.500	0.429	3	6
Subject 8	0.59	0.36	6.5	1.5	0.667	0.500	8	2

Table 1: Average correct ($\overline{p}$, $\overline{i}$) and total correct ($k\overline{i}$, $k\overline{i}$) observations per subject

Command	Condition	% Total	Mean	SD	Median	IQR
TotalOperations	User+Aide		331	158	361	297
	UserAlone		191	83	180	144
LocalOperations	User+Aide	38%	127	84	118	134
	UserAlone	73%	140	81	143	122
NavigationOperations	User+Aide	44%	146	73	137	148
	UserAlone	12%	22	5	21	9
ManipulationOperations	User+Aide	13%	44	37	28	65
	UserAlone	9%	17	21	9	24

Table 2: Summary of operations selected, averaged over all subjects

the performance improvement; in other words, better performance in the User+Aide condition is not due to subjects simply seeing more of the data. It is also not the case that subjects in the UserAlone condition never happen upon the relationships and patterns suggested by Aide in the User+Aide condition. Of all the correct suggestions Aide made about each dataset, only one was not also tried by subjects in the UserAlone condition.

Explaining subject performance

Having established that subjects explore data better with Aide than without, we can take on the more difficult problem of explaining Aide's contribution to the process. This will involve extracting patterns from the traces of subject behavior and relating the patterns to performance results (using, for the most part, Aide itself to carry out the analysis.)

Exploratory operations in Aide can be divided roughly into three types. Some operations are concerned with local decision-making: selecting a variable or constructing a relationship for display, examining indications, or asking the system for documentation of proposed actions. These are what we will call LocalOperations. They involve decision-making at a single focus point: assessing information about which variables and relationships it would be worthwhile to describe, or evaluating the applicability of different operations and procedures to describe a potential pattern. LocalOperations account for 40% of the operations in the User+Aide condition. NavigationOperations are such actions as initiating the exploration of a variable or relationship or going back after generating a result to select another relationship. In other words, these operations generate new focus points, or take the exploration from one focus point to another. Navigation is responsible for almost half (44%) of the operations in the User+Aide condition. Finally, ManipulationOperations are a specific type of navigation operation, involving selection of the reductions, transformations, and decompositions that make changes or additions to the data. Data manipulation accounts for only a small portion of the total number of operations. Table 2 gives a statistical summary of the operations made in each condition for all subjects. Because the distributions are somewhat skewed, the table presents the median and interquartile range as well as the mean and standard deviation. Statistics for the UserAlone condition are given for comparison.

These summaries give us a rough idea of how subjects went about exploring a dataset. Much of the effort, in terms of the number of operations applied, involved examining the data from different angles and evaluating ways of building descriptions. Subjects showed a good deal of mobility, not just in moving from one data structure to the next, but also in moving from one point in the network of exploratory plans and actions to another. This point was also emphasized by most of the subjects in their assessments: a common theme was the importance of being able to navigate through the exploration process. The summaries also show that data manipulation was secondary to other activities; we might even think of navigation and local evaluation of decisions as setting the stage for data manipulation.

The difference between the User+Aide and UserAlone conditions is striking. While local decision-making is the most important factor in the UserAlone condition, navigation dominates in the User+Aide condition. We infer that the navigational facility, which relies on an explicit model of the data analysis process, above the level of individual operations, is a factor in improved performance in the User+Aide condition.

These variables are related as shown in the model in Figure 1, produced by Aide's causal modeling procedures. In this model and the ones that follow, nodes represent features of the system (in the User+Aide condition), and arcs represent causal influences between them. These models were constructed using causal modeling techniques comparable to those in the literature, such as Pearl's IC (Pearl & Verma 1991) and Spirtes et al.'s PC (Spirtes, Glymour, & Scheines 1993), modified when necessary to reflect our understanding of the variables and relationships involved (Cohen 1995). We have eliminated TotalOperations from this model, because, as we know, it is a linear function of the other three variables. The relationship between NavigationOperations and LocalOperations is relatively strong ($r = 0.67$), as is the relationship between NavigationOperations and ManipulationOperations ($r = 0.54$). The variables ManipulationOperations and LocalOperations are weakly correlated to begin with ($r = 0.29$), and if we hold NavigationOperations constant the correlation drops to 0.12. Exploration of the relationships in this model shows no unusual patterns.

We can also connect our modeling efforts with the results of the last section by bringing performance measures into the picture, as shown in Figure 2. As expected, the identification of a relationship influences its correct description (i.e. $\overline{i}$ influences $\overline{p}$.) Otherwise, $\overline{i}$ is influenced by NavigationOperations, while both $\overline{p}$ and $\overline{i}$ are influenced by ManipulationOperations. This latter set of influences is interesting, in that explicit data manipulation accounts for relatively little of the total effort a subject puts into the exploration, but is a factor in determining both performance measures. A plausible explanation is that data manipulation operations are generally applied only when one perceives some kind of pattern. Data manipulation operations generally provide a more detailed view of a pattern, and thus a greater number of these operations leads to more accurate observations. This model gives us a set of plausible, tentative notions about how subject behavior, in terms of the types of operations applied, influences performance.

Explaining AIDE's behavior

Up to this point our analysis has concentrated on the behavior and performance of individual subjects. We want to bring in other factors that influence performance, such as the type of pattern involved in a description, but we run into a problem: this kind of property characterizes observations, not subjects. It makes little sense to compute some kind of pattern-type score for each subject, averaging over all observations.

Let's take a different tack. If we are willing to aggregate the results of all the subjects, we can build a tentative model of Aide's behavior based on the properties of individual observations, rather than of individual subjects. As before, we concentrate on the observations made during the User+Aide condition. We can now use properties of individual observations as variables:

- O_p: whether an observation correctly describes a

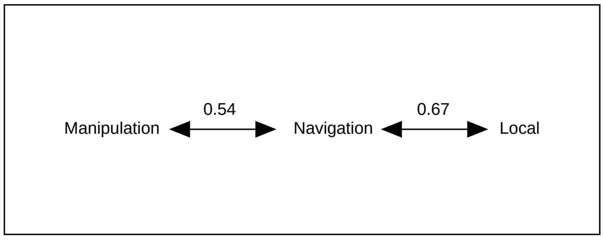

Figure 1: Model of operation counts

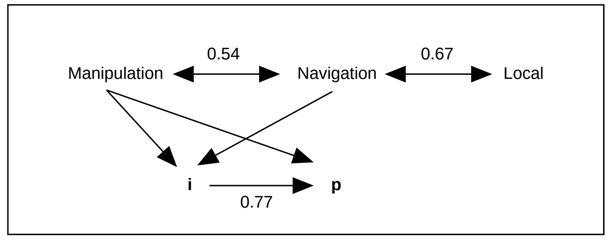

Figure 2: Model of operation counts, plus performance

variable or direct relationship in the data. O_p is T if correct, F otherwise.

- O_i: whether an observation identifies a direct relationship in the data. O_i is T if direct, F otherwise.
- *Suggested:* whether AIDE suggested a variable or relationship for exploration. A common theme in subject assessments was the helpfulness of AIDE's suggestions in imposing structure on the exploration. For some subjects this even led to differences in when observations were made, over the life of a trial: in the USER+AIDE condition most observations were made early on, toward the beginning of the trial, while in the USERALONE condition they were more evenly distributed over the entire session.
- *Pattern type:* the type of pattern identified. Because AIDE has different plans to suggest and generate descriptions of different types of patterns, we expect to see some interaction between this variable and our performance measures.

We might think that AIDE's performance could be directly determined from the performance of individual subjects. That is, we might think that when subjects subjects do well, AIDE is doing well, and when subjects do badly, this is AIDE's fault as well. However, performance cannot be so simply decomposed. Subjects often rejected AIDE's suggestions, and tried more possibilities on their own initiative. The resulting observations thus depend not only on AIDE's efforts, but also on whether subjects accepted AIDE's advice. Overall we see a dependence between Suggested and O_i, as shown below; a χ^2 test gives a result of 3.298, with a significance of 0.069. AIDE's suggestions are in general effective.

	$\overline{\text{Suggested}}$	Suggested	Totals
$O_i = F$	14	18	32
$O_i = T$	14	42	56
Totals	28	60	88

Clearly there will be a strong relationship between O_i and O_p, as can be seen below. Any relationship that is correctly described must first be correctly identified, which imposes a strong dependence between the two types of correct observations, as well as incorrect ones. Between these variables we find a χ^2 of 47.665, which is extremely significant.

	$O_p = F$	$O_p = T$	Totals
$O_i = F$	31	0	31
$O_i = T$	12	42	54
Totals	43	42	85

Now, AIDE's initial suggestions, concerning which data structures to explore, are based on activation and preference rules that consider the types of patterns present in the data when making their decisions. Thus we might expect to see a relationship between Suggested and O_p, but we find none. This is actually not surprising. Suggested and O_p can only be indirectly related to one another; patterns are only potentially identified by suggestions, to be extracted through plans that are executed later. The decisions made in these plans at the lateter stages will have a much stronger influence on the correctness of the resulting descriptions.

The variable Pattern-Type interacts with both O_i and O_p. That is, the likelihood that a pattern is identified and described correctly depends on its type. This is plausible, because activation and preference rules depend on patterns in the data. Thus Pattern-Type and O_i should be related. Because different plans are used to generate different types of patterns, Pattern-Type and O_p are also related, to an even greater extent. The final model is shown in Figure 3. Further exploration of the data turned up no significant additional information.

Discussion

AIDE differs from conventional statistical interfaces in maintaining an explicit model of the *process* of exploration, in addition to its individual operations. Thus the user can tell AIDE, "Back up a step in the analysis," or "Show me the decisions leading to the current state," or "Stop doing x; consider decision y instead." The breakdown of operations by type showed that navigation was in fact the most common type of operation during exploration. Our results illustrate the benefit of viewing interaction as an exchange in a persistent procedural context—a planning context—rather than as a series of one-shot transactions (Ferguson, Allen, & Miller 1996). For autonomous data exploration systems that concentrate on local decisions in a static modeling framework (e.g., deciding whether to split a continuous variable in a decision tree, or deciding

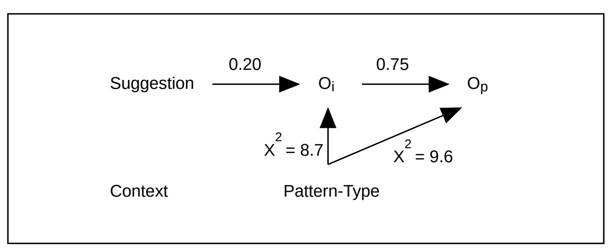

Figure 3: Model of performance relationships

whether to add a variable to a regression model), one implication is that much can be gained by increasing flexibility and opportunism at a more global level.

AIDE owes a debt to earlier systems developed to explore the concepts of statistical strategy. These include REX, an expert system for regression analysis (Gale 1986), TESS, a search-based system for data description(Lubinsky & Pregibon 1988), and systems by Hand (Hand 1986) and Oldford and Peters (Oldford & Peters 1986). Gale, Hand, and Kelly describe other relevant work in an excellent comprehensive review (Gale, Hand, & Kelly 1993).

From an AI standpoint, AIDE is an example of a collaborative system (Grosz 1996). Collaborative systems generally face a number of difficult problems, including being able to reason about shared goals and context, deciding how to allocate responsibilities, and communicating about coordination and evaluation (Terveen 1995). Some of these problems are alleviated in AIDE by the partial hierarchical planning approach and the constraints of the domain. For example, as a general purpose system AIDE is designed to have no knowledge of subject-matter context—what the data mean. This puts limitations on the degree of strategic decision-making that AIDE can exercise. Instead, the user generally remains in charge of deciding where the exploration should go, while AIDE attempts to takes on the tactical burden of exploration.

In summary, AIDE helps users to make decisions about the presence of structure in the data, better decisions than are made without AIDE's help. AIDE improved the performance (by different measures) of almost all the subjects in the experiment. Data manipulation and navigation operations turn out to be the most important factors in determining performance, though actual data manipulation accounts for only a small proportion of the total activity. Other relevant factors are the quality of the suggestions AIDE makes to the user and the type of result, or description, being generated. AIDE has definite limitations, especially in the depth of its plan library and sophistication of its internal planner. Nevertheless, our experimental findings are encouraging evidence that AIDE is a step toward collaborative human/computer data analysis.

Acknowledgments

Thanks to anonymous reviewers for suggestions about the direction this paper should take. This research is supported by DARPA/Rome Laboratory under contract #F30602-93-C-0076, and by a DARPA/ARO Augmentation Award for Science and Engineering Research Training (AASERT) under contract #34664-RT-AAS. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarioy representing the official policies or endorsements either expressed or implied, of the Defense Advanced Research Projects Agency, Rome Laboratory or the U.S. Government.

References

Allen, J. F. 1994. Mixed initiative planning: Position paper. [WWW document]. Presented at the ARPA/Rome Labs Planning Initiative Workshop. http://www.cs.rochester.edu/research/trains/mip/.

Carver, N., and Lesser, V. 1993. A planner for the control of problem solving systems. *IEEE Transactions on Systems, Man, and Cybernetics, special issue on Planning, Scheduling, and Control* 23(6):1519–1536.

Cohen, P. R. 1995. *Empirical Methods for Artificial Intelligence.* MIT Press.

Ferguson, G.; Allen, J.; and Miller, B. 1996. TRAINS-95: Towards a mixed-initiative planning assistant. In *Proceedings of the Third International Conference on Artificial Intelligence Planning Systems*, 70–77.

Gale, W. A.; Hand, D. J.; and Kelly, A. E. 1993. Statistical applications of artificial intelligence. In Rao, C. R., ed., *Handbook of Statistics*, volume 9. Elsevier Science. chapter 16, 535–576.

Gale, W. A. 1986. REX review. In Gale, W. A., ed., *Artificial Intelligence and Statistics I.* Addison-Wesley Publishing Company.

Georgeff, M. P., and Lansky, A. L. 1986. Procedural knowledge. *Proceedings of the IEEE Special Issue on Knowledge Representation* 74(10):1383–1398.

Grosz, B. 1996. Collaborative systems. *AI Magazine* 17(2).

Hand, D. 1986. Patterns in statistical strategy. In Gale, W., ed., *Artificial Intelligence and Statistics I.* Addison-Wesley Publishing Company. 355–387.

Lubinsky, D., and Pregibon, D. 1988. Data analysis as search. *Journal of Econometrics* 38:247–268.

Oldford, R. W., and Peters, S. C. 1986. Implementation and study of statistical strategy. In Gale, W., ed., *Artificial Intelligence and Statistics I.* Addison-Wesley Publishing Company. 335–349.

Pearl, J., and Verma, T. 1991. A theory of inferred causation. In *Proceedings of Principles of Knowl-*

edge Representation and Reasoning, 441–452. Morgan Kaufmann Publishers, Inc.

Spirtes, P.; Glymour, C.; and Scheines, R. 1993. *Causation, Prediction, and Search.* Springer-Verlag.

St. Amant, R., and Cohen, P. R. 1996a. Control representation in an EDA assistant. In Fisher, D., and Lenz, H., eds., *Learning from Data: AI and Statistics V.* Springer-Verlag. 353–362.

St. Amant, R., and Cohen, P. R. 1996b. A planner for exploratory data analysis. In *Proceedings of the Third International Conference on Artificial Intelligence Planning Systems*, 205–212. AAAI Press.

St. Amant, R., and Cohen, P. R. 1997. Interaction with a mixed-initiative system for exploratory data analysis. In *International Conference on Intelligent User Interfaces.*

St. Amant, R. 1996. *A Mixed-Initiative Planning Approach to Exploratory Data Analysis.* Ph.D. Dissertation, University of Massachusetts, Amherst. Also available as technical report CMPSCI-96-33.

Terveen, L. G. 1995. An overview of human-computer collaboration. *Knowledge-Based Systems* 8(2–3):67–81.

Tukey, J. W. 1977. *Exploratory Data Analysis.* Addison-Wesley Publishing Company.