Building an EDA Assistant: A Progress Report

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

Keywords: automated data analysis, integrated man-machine methods

Since 1993 we have been working on a system to help people with exploratory data analysis (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. Its plan library contains strategies for generating and interpreting indications in data, selecting techniques to build appropriate descriptions of data, carrying out relevant procedures, and combining individual results into a coherent larger picture. The system is mixed-initiative, autonomously pursuing high- and low-level goals while still allowing the user to inform or override its decisions.

Elsewhere we have described AIDE's operations and primitive data structures [22], its planning representation [23], its user interface [25, 24], and the system as a whole [21]. This progress report discusses a recent evaluation we conducted with AIDE and explains why we believe that this line of research is important to AI and statistics researchers.

We will begin with a very brief overview of the system. The bulk of the paper describes the evaluation, our analysis of the results, and the lessons we learned through the experience of building and evaluating AIDE. We end with a discussion of the generality of our results and the potential for future work.

1 AIDE and Planning

AIDE's design exploits a striking similarity between interactive data exploration and a type of AI planning known as partial hierarchical planning [22, 23]. AIDE maintains a library of over a hundred plans and control rules representing knowledge about how statistical procedures are carried out. Each plan is designed to capture an element of common statistical practice, such as the examination of residuals after fitting a function to a relationship, the search for refinements and predictive factors when observing clustering, the application of various data reduction techniques, and so forth. The plans lack human-level knowledge of subject-matter context—what the data actually mean—but the selection of each action is sensitive to the procedural context generated by the actions that have gone before.

AIDE explores as follows. When a dataset or relationship is presented to the system, a goal is established for its exploration. To satisfy this goal, the system searches through its library for an appropriate plan. This plan is expanded into its component subgoals, which are satisfied in turn. The expansion bottoms out when a goal is satisfied by the retrieval of a primitive action, which

contains directly executable code, from the library. When a goal can be satisfied by more than one plan or action in the library, a *focus point* is created to handle the choice between them; the system chooses one of the possibilities to continue. In the course of the exploration, AIDE may uncover *indications,* or suggestive features in the data [13], that cause it to reconsider some of its earlier decisions. When this happens, it can return to a relevant focus point, modify the original decision, and continue from that point.

The user interacts with AIDE through an extended statistical interface that also provides access to conventional statistical tools. 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 each of the user's actions can be recorded and potentially interpreted by the system.

Mixed-initiative interaction is central to AIDE's design. The goal of combining a partially autonomous planner and an accommodating statistical interface is to relieve the user of the routine or search-intensive aspects of exploration, without precluding human guidance of the entire process. Thus while the system can make decisions alone, the user can take control at any point to review and modify its choices. Conversely, even when the user explicitly selects each action, the system's proposed actions offer advice about how to proceed as each new result is generated. AIDE's focus point network ties the entire process together to give an explicit, structured justification for decisions and results. This structure provides the basis for a useful metaphor of exploration as navigation through a space of statistical decisions.

2 Evaluation

Evaluation focused on a simple hypothesis:

Exploration is more effective with AIDE *than without.*

Our main goal was to demonstrate empirically that the strategies developed in AIDE could be put to good use. Two issues are tied up in this goal, usability and performance. That is, we want to show both that users find the system helpful and that results obtained with AIDE are better than those obtained without its help.

We also considered two subsidiary issues related to the human element in exploration. One danger in increasing the autonomy of a system is that its less predictable behavior may cause users to mistrust its results. In contrast, an ideal assistant would cause users to be more confident in results generated with its help. Phrased as a hypothesis,

AIDE*'s assistance improves user confidence in results.*

In our evaluation this turned out not to be the case, with one suggestive exception. Fortunately, however, neither did the converse hold: AIDE's participation did not reduce confidence.

Much of the literature in statistical expert systems (and collaborative systems in general) holds that an important factor in successful exploration is the contextual knowledge, or knowledge of what the data mean, that a user brings to bear [9, 12, 27]. Our experiment attempted to quantify this factor and determine its influence on exploration. In other words, we wished to show that

The presence of contextual knowledge influences the effectiveness of exploration.

Exploring this issue may give us clues about when and how AIDE provides benefits in exploration. It could happen that AIDE compensates for a lack of contextual knowledge, giving better performance

for situations in which a user knows little about the variables and relationships in a dataset. Alternatively, it could happen that AIDE works better when the user has some contextual knowledge, so that the reverse holds for performance.

2.1 Experiment design

The experiment involved testing subjects under two conditions. In the USER+AIDE condition, subjects explored a dataset with AIDE's help, while in the USERALONE condition, subjects explored a dataset in a similar statistical computing environment but without active help from AIDE. AIDE's effectiveness was then determined by measuring differences in performance between the two conditions. The design is a straightforward matched-pair comparison, arrived at through consideration of several potentially confounding factors.

Extraneous variability in conditions: In order to prevent usability issues from entering the picture, we must ensure that the system used in the USER+AIDE condition is as close as possible to the system in the USERALONE condition. That is, we cannot simply use AIDE in the USER+AIDE condition and, say, JMP [20] or CLASP [2] in the USERALONE condition, because differences in performance would then not necessarily be attributable to AIDE, but perhaps to differences in styles of user interaction. To avoid this problem, the USERALONE condition reproduces AIDE's environment, lacking only the intelligent interaction capabilities. For example, in the USER+AIDE condition, the system might detect a comparatively high correlation and suggest that the relationship (x, y) be explored, then propose and execute a linear fit procedure on the data, exploring residuals afterwards. In the USERALONE condition, the user receives no suggestions about which relationships might be worth exploring; once the user selects the relationship (x, y), the system gives no advice about how to describe the data; the user must explicitly select a regression or resistant fit for the data, and then explore the residuals with further explicit commands. In the USER+AIDE condition, AIDE takes over some of the control, with the accompanying possibility of leading the exploration astray, while in the USERALONE condition the user must select every action.

Variability in subjects: Suppose we divided subjects into two groups, Group A to explore a dataset in the USER+AIDE condition, Group B to explore the same dataset in the USERALONE condition. Suppose further that Group A subjects outperformed Group B subjects, on average, for some performance measure. Unfortunately, because different subjects have different facility with EDA techniques, the performance of Group A might be attributable to random variation in user ability.

The paired comparison design removes this variability. Each subject is tested in both conditions. Our performance measures are then not biased by differences in ability between individual subjects, because these abilities will be represented in both conditions. We can compare performance differences for individuals, as well as aggregate measures of performance per condition.

Practice effects and variability in problem difficulty: If subjects are to be tested in both conditions, we immediately face a serious practice effect. A subject who has explored a dataset in one condition will certainly be able to take advantage of this knowledge in the other condition. Randomizing the presentation of conditions is no help in this case. Giving subjects two different datasets may also pose problems: how can we compare performance between them? The datasets might be very different in their structural characteristics and the patterns they contain.

Artificial data is the answer. We can generate datasets based on identical or nearly identical models that give us datasets with very similar characteristics. In doing this we need to be careful that the information gained from exploring one dataset does not help in exploring another. This turns out not to be a problem; knowing that a log relationship holds between x and y in dataset

D_1, for example, does not help us to identify and describe a similar relationship in dataset D_2. Using artificial data also gives us the opportunity to manipulate the level of contextual knowledge available to users. We give some variables plausible names that correspond to natural relationships with the other variables, and others anonymous names like "v1" and "v2".

Ordering effects: In pilot runs of the experiment we found that each phase of each trial (i.e. testing in each condition) took from one to two hours. This potentially gives rise to an ordering effect: the subject may become more comfortable with the task or the system during the first phase, and have less difficulty during the second.

A counterbalanced design controls for this effect. The order in which the subject is presented with the two conditions is randomized. Thus even if subjects *always* do better in the second phase, the improvements will apply to both the USER+AIDE and the USERALONE conditions. (The latter possibility turned out not to be the case, as we found in the analysis of the experimental data.)

Variability in effort: We have designed AIDE to help the user explore a larger search space than he or she might explore alone. One might then expect a user, given sufficient time, to be able to find all the structure AIDE would find. Under practical experimental conditions, however, we can't give the subject unlimited time to perform the exploration. On the other hand, if we give subjects too little time, we may be biasing the experiment in favor of AIDE, in the same way computers have an advantage playing blitz chess.

Here we decided to rely on the subject's judgment about when the task is complete. We had no reason to believe that this judgment would differ in the two conditions. The subject receives instructions to present all results he or she considers significant in summarizing or explaining significant structure in the dataset. (During the experiment, subjects ended up spending about the same amount of time in each condition.)

To summarize, 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 dataset/condition assignment was randomized, as was the order in which the datasets were explored. Subjects were instructed to make notations identifying and describing the direct relationships between variables in the data they explored. Subjects also annotated each description with a confidence rating, "high" or "low", indicating how confident they were that their judgment was correct. Because of the time and effort involved in overseeing individual trials, which lasted on the order of four hours each, the experiment was limited to eight subjects.

2.2 Test data

Two artificial datasets were generated with specific criteria in mind. They should contain patterns and structural relationships amenable to exploration. These include linear and clustering relationships between variables, functional relationships between variables that depend on the values of other variables, and so on. The two datasets should furthermore be similar, in the sense that exploring one should be no harder or easier than exploring the other. Nevertheless information gained in exploring one dataset should not help in exploring the other.

The generation of each 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. Nodes with no incoming arcs, or exogenous nodes, are associated with specific distributions. A row of the dataset is computed 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 rows

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.

The names of the variables in the generated datasets were carefully chosen to appear meaningful, but also to give no indication of direct causal relationship or causal order. The model for each dataset consists of two subgraphs (g_1 and g_2) connected at two nodes. The two models are essentially the same, except for the ordering of their subgraphs. To disguise the similarities between the models, variables in subgraph g_1 of one model and subgraph g_2 of the other model have anonymous names. From a subject's point of view, each model consists of named and anonymous variables, and there is no overlap in naming between the models.

It is important to note that in no sense was AIDE tuned to the specific patterns in these datasets. The system developers had no role in building the data generator and producing the datasets, and there was no contact with the data before the experiment with AIDE began.

2.3 Results

We defined several related measures of performance: the average number of direct relationships correctly identified and described, over all subject notations made ($\overline{p}$); the total number of correct notations ($k\overline{p}$); the average number of direct relationships correctly identified, without regard to their correct description ($\overline{i}$); the total number of correct identifications ($k\overline{i}$). The measures $\overline{i}$ and $k\overline{i}$ deal with the connectivity of a causal model, while $\overline{p}$ and $k\overline{p}$ address its descriptive annotation.

Subjects performed as shown in Table 1. A matched-pair, one-tailed t-test tells us that $\overline{p}$ and $k\overline{p}$ are significantly higher for subjects in the USER+AIDE condition: $t = 2.217$ and 1.808, with p-values around 0.03 and 0.05, respectively. A similar result holds true for $\overline{i}$ and $k\overline{i}$.

The 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. That is, given our experimental conditions, users can perform EDA better with the help of AIDE than they can alone, and better than AIDE acting alone.

To put this comparison in perspective, we will consider a few plausible 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 two 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. Better performance is not due to subjects simply seeing more of the data in the USER+AIDE condition.

	$\overline{p}$		$k\overline{p}$		$\overline{i}$		$k\overline{i}$	
	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

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 considered by subjects in the UserAlone condition.

Let's move to the second hypothesis. How does Aide affect the confidence of subjects in the results they produce? If we combine confidence values (taking "high" as 1, "low" as 0) for all observations made by each subject, we arrive at a measure of the confidence of a subject during each condition. The mean confidence C_M of subjects in the User+Aide condition ($C_M = 0.599$) turns out to be not significantly different from that of subjects in the UserAlone condition ($C_M = 0.628$). This raises an obvious question of whether subjects have different confidence in observations that turn out to be correct than they do for incorrect observations. In fact, this is an important point: we are happy if a system makes subjects confident in their activities, but not if their results turn out to be consistently wrong. The results are shown in Table 2. When we break the dataset down into correct and incorrect observations, both for correct description ($\overline{p}$) and correct identification ($\overline{i}$), we find that confidence is higher for correct observations than for incorrect ones. The general pattern is the same for both measures of performance, with one suggestive exception. Confidence levels for correct observations are about the same in both conditions, but for incorrect observations (measured by $\overline{i}$) confidence levels are noticeably lower in the User+Aide condition. That is, for one interesting subset of cases subjects have more appropriate confidence levels in the User+Aide condition than in the UserAlone condition.

These results are equivocal. The good news is that Aide does not reduce user confidence, a common occurrence and thus a serious concern for intelligent assistants [19]. On the other hand, we cannot thereby conclude that Aide has a positive, balancing effect on user confidence; as one of the experimental subjects suggested, confidence may depend on factors independent of Aide's autonomous activities. Further work in this area is needed.

The third issue concerns the effect of contextual knowledge on performance. An analysis of variance examined the interaction between the condition (User+Aide orUserAlone) and the presence or absence of contextual information for observations, in the form of meaningful names for variables. The analysis was somewhat involved; we will simply summarize by saying that contextual cues in the data were not strong enough to lead subjects directly to correct descriptions (as measured by $\overline{p}$), but nevertheless point in the right direction by drawing attention to those relationships worth pursuing (as shown by $\overline{i}$). This result is suggestive and intuitively plausible.

2.4 Explaining Subject Performance

Now, the simple fact of a performance difference between the User+Aide and UserAlone conditions is not entirely satisfying. We are really most interested in understanding why Aide works. For a better view of Aide's contribution, we divided subject actions 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.

	$C_M(\overline{p})$		$C_M(\overline{i})$	
	Aide	Alone	Aide	Alone
Correct Means	0.68	0.69	0.66	0.66
Incorrect Means	0.56	0.58	0.48	0.56

Table 2: C_M per subject, for correct and incorrect observations

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 3 gives a 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.

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. Examining the relationships in more detail, we find that 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 these relationships shows no unusual patterns. In relating these factors to our performance measurements, it appears that explicit data manipulation accounts for relatively little of the total effort a subject puts into the exploration, but is one of the strongest factors in determining performance. Navigation is the other important factor. 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.

Though our understanding of these factors is very tentative, they 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.

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 3: Summary of operations selected, averaged over all subjects

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 think of navigation and local evaluation of decisions as setting the stage for data manipulation.

2.5 Limitations

These experimental results are a promising first step, but their generality is limited in several ways.

First, and most obviously, time and resource constraints left us with a very small sample size. The effects were large enough to see even with only eight subjects, but an experiment on a larger scale might have helped us make more headway in our investigations into user confidence and context. Our current findings provide a useful pilot study, however, for further tests with a larger group of subjects.

Second, we used specific artificial datasets to provide a necessary experimental control. These datasets were constructed to reflect realistic, common patterns; nevertheless, one might ask how AIDE would perform on other datasets that contain patterns and relationships not present in the test data. Because AIDE's strategies were developed to handle patterns in a variety of datasets [31, 3], we are confident in AIDE's robustness. Empirically establishing this robustness is a difficult problem, however, and will need further work.

A third point is also related to the use of artificial data: are AIDE's strategies up to the requirements of real world problems? In informal testing on real datasets, we found AIDE to be helpful for specific types of patterns. In general, however, this is another open question, requiring further development and experimentation.

3 Discussion

Here are some of the lessons we learned in developing and evaluating AIDE. Our findings are largely consistent with other reviews of statistical expert system development [17, 8, 30, 18, 6, 16], but we also identify a few fresh directions for research. We will begin with the research questions we addressed.

Planning is a practical means of supporting the data analysis process. Note the inclusion of the word "process". Data analysis is different from, for example, word processing and batch programming: the correctness of the end product cannot be checked without inspecting the path leading to it [10, p. 69]. A great deal of work in statistical strategy takes this view. The most prominent example is probably Gale and Pregibon's REX system, which implemented a strategy for linear regression [5]. REX's actions were determined by the traversal of a decision tree; the tree provides an explicit representation of the sequential, coherent decision-making process. In contrast, conventional software for data analysis focuses on powerful individual operations, or a comfortable statistical programming environment, but provides little support for the structured organization of these operations and procedures.

AIDE supports the data analysis process more directly. Imagine in the course of exploring a dataset you decide to build a linear model of a set of variables. During the process you notice an unusual pattern of clusters in a subset of the data, and you suspend your modeling to follow this tangent. When you are finished, you return to the point at which you broke off, to continue with the model. By maintaining an explicit representation of the exploration *process*, in addition to its individual actions, AIDE can support this kind of navigation. AIDE furthermore helps to reorient the user in making such shifts in attention, by presenting the chain of decisions leading

to a given point, displaying relevant data, making appropriate suggestions—in general, helping to restore context, as far as possible given the built-in limitation of the system's knowledge.

Shared control is a key aspect of effective assistance. The perspective we take with AIDE is that human involvement is an essential part of the exploratory process. A completely autonomous system can have little notion of the significance of its findings—but this is exactly the kind of knowledge that informs the selection of data, analysis methods, and evaluation techniques. We view exploratory systems as trading off autonomy and accommodation, where "accommodation" means a responsiveness to knowledgeable human guidance [12].

AIDE balances autonomy and accommodation within the framework of a partial hierarchical planner, which generates an explicit representation of the exploration process for the user's review and potential modification. As a mixed-initiative planner [1], and also as a collaborative system [28], AIDE must *assist* in an exploration, rather than taking it over completely or waiting for instructions for each of its moves. This approach has several benefits, one of the most important being its flexibility. For example, a mixed-initiative system can potentially be acceptable to both novices and experts. A common problem faced by an intelligent assistant—in fact, by most user interfaces—is that providing comprehensive guidance and support for novice users can actively impede expert users. Conversely, building systems to support experts may entail an enormous learning curve for novices. In AIDE's mixed-initiative design, the system offers advice and analysis paths which may be helpful for novice users, but its decisions can be overridden at almost any point by an expert user. The interaction is not perfect for an expert user. For example, AIDE may consider decisions in a different order from the expert, who will have to guide the analysis at each step, occasionally rolling the analysis back to an earlier state if AIDE jumps ahead. Nevertheless, this kind of interaction has been made as easy as possible, for just such cases.

Maintaining context can be difficult problem in a mixed-initiative system. Sharing control is not without pitfalls. Whenever AIDE takes control of the analysis, it runs the risk of losing the user. This problem applies to many domains other than statistical analysis; in interaction with hypertext systems, for example, it is called the "lost in hyperspace" feeling [14].[1] This is a basic human-computer interaction concern: the system should provide the user with implicit answers to the questions, "Where am I?" "How did I get here?" "What can I do here?" and "Where can I go from here?" [15]. These are exactly the questions handled by AIDE's navigation facilities.

The experience of building a statistical assistant yielded some additional practical lessons:

Building an intelligent assistant as an independent agent can impose constraints on its abilities. Modern intelligent statistical systems are commonly designed as front-ends to existing statistical packages; for example, ViSta [32] and Omega-Stat [11] are built on top of xlisp-stat [29]. In contrast, some early statistical expert systems were built outside an existing statistical package, designed for only loosely coupled interaction [5, 6]. If the resulting system can use the statistics package through the same interface as the user, the arrangement has some strong advantages: the agent and user share the same vocabulary of actions; much of the agent's step-by-step reasoning can be displayed directly through the interface; strategy acquisition on the part of the agent (or programming by demonstration) is facilitated.

Unfortunately, an automated agent has requirements and abilities very different from those of a human user. For example, an agent can accurately store and retrieve large amounts of data without reminders; on the other hand, it has no visual pattern processing abilities—the large number of

[1] Hand's knowledge enhancement system, KENS, let users browse through a network structure of statistical concepts containing over 200 nodes [7]. KENS is similar in some ways to hypertext systems, but users of KENS had no problems orienting themselves—a surprising and significant result.

functions in modern statistics packages devoted to graphical displays are wasted on a system like AIDE. Most of the functionality of an interactive statistics package, whether in its graphics or its programming language, is geared toward human ease of use and understanding. For AIDE, this meant relaxing the strict separation between itself and its statistics package, CLASP [2], to avoid reimplementation of needed functionality. While the loose coupling has advantages, there are costs as well.

Evaluation must be considered from the start. One of the difficulties we faced in evaluating AIDE was ensuring that user interface issues did not confound our results. Evaluation of intelligent interactive systems is relatively uncommon. For example, in a collection of papers from the first conference on intelligent user interfaces [26], sixteen implemented systems are described; only two of these descriptions contain a discussion of empirical evaluation of the work. We speculate that one reason for this (besides unfamiliarity with empirical methods [4]) is a lack of foresight in system development. One can easily build a system that accomplishes a task in a completely new way; demonstrating that it is an improvement on existing methods is sometimes a much more difficult problem.

In order to show that AIDE improves on existing interfaces, we needed to establish a baseline for comparison. That is, we continually reminded ourselves that the final evaluation must be a fair comparison between new functionality and old. This meant, for example, working on two versions of the system, one for the USER+AIDE condition and the other for the USERALONE condition. Both needed adequate functionality for the test to be fair. If we had not considered this problem early in the development of the system, it would have been very difficult to tease apart the factors that might account for differences in performance.

4 Conclusion

In an insightful review of progress in building statistical expert systems, Gale, Hand, and Kelly [6] note that few of these systems have had a significant impact on conventional statistical computing or commercial software. Most systems lived only as research prototypes, never reaching wide use or commercial feasibility. It is fair to ask then what we have learned that makes us optimistic about the future of statistical expert systems like AIDE.

First, our planning design for AIDE has given us a useful way to interpret the EDA process. Mixed-initiative planning is an active area of research in the AI planning community, and its techniques hold a great deal of promise in other domains. As progress is made on basic research issues we can incorporate new ideas into the AIDE framework.

Second, we believe that an empirical approach to building systems like AIDE is the most effective way of making progress. Our experimentation with AIDE has identified some clear directions for further work, especially in the areas of navigation and strategy visualization. These and other areas often come to be noticed only by their appearance in empirical studies and their influence on results.

5 Acknowledgments

Thanks to anonymous reviewers who offered excellent suggestions about the direction this paper should take. This research is supported by ARPA/Rome Laboratory under contracts F30602-91-C-0076 and F30602-93-C-0010. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes not withstanding any copyright notation hereon.

References

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

[2] Scott D. Anderson, David M. Hart, David L. Westbrook, , and Paul R. Cohen. A toolbox for analyzing programs. *International Journal of Artificial Intelligence Tools*, 4(1 & 2):257–279, 1995.

[3] D.F. Andrews and A.M. Herzberg. *Data: A Collection of Problems from Many Fields for the Student and Research Worker*. Springer, 1985.

[4] Paul R. Cohen. *Empirical Methods for Artificial Intelligence*. MIT Press, 1995.

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

[6] William A. Gale, David J. Hand, and Anthony E. Kelly. Statistical applications of artificial intelligence. In C. R. Rao, editor, *Handbook of Statistics*, volume 9, chapter 16, pages 535–576. Elsevier Science, 1993.

[7] D. J. Hand. A statistical knowledge enhancement system. *Journal of the Royal Statistical Society Serial A*, 150:334–345, 1987.

[8] D.J. Hand. Patterns in statistical strategy. In W.A. Gale, editor, *Artificial Intelligence and Statistics I*, pages 355–387. Addison-Wesley Publishing Company, 1986.

[9] Peter J. Huber. Data analysis implications for command language design. In K. Hopper and I. A. Newman, editors, *Foundation for Human-Computer Communication*. Elsevier Science Publishers, 1986.

[10] Peter J. Huber. Languages for statistics and data analysis. In Peter Dirschedl and Ruediger Ostermann, editors, *Computational Statistics*. Springer-Verlag, 1994.

[11] E. James Jarner and Hanga C. Galfalvy. Omega-Stat: An environment for implementing intelligent modeling strategies. In Douglas Fisher and Hans Lenz, editors, *Learning from Data: AI and Statistics V*, pages 333–342. Springer-Verlag, 1996.

[12] David Lubinsky and Daryl Pregibon. Data analysis as search. *Journal of Econometrics*, 38:247–268, 1988.

[13] Frederick Mosteller and John W. Tukey. *Data Analysis and Regression*. Addison-Wesley Publishing Company, 1977.

[14] Jakob Nielsen. *Hypertext and hypermedia*. Academic Press, Inc., 1990.

[15] J. Nievergelt and J. Weydert. Sites, modes and trails: Telling the user of an interactive system where he is, what he can do, and how to get to places. In Ronald M. Baecker and William A. S. Buxton, editors, *Readings in Human-Computer Interaction: A Multidisciplinary Approach*, pages 438–441. Morgan Kaufmann, 1987.

[16] Carl M. O'Brien. Are there any lessons to be learnt from the building of glimpse? In D. J. Hand, editor, *AI and Computing Power*, pages 53–62. Chapman & Hall, 1994.

[17] Daryl Pregibon. A diy guide to statistical strategy. In W.A. Gale, editor, *Artificial Intelligence and Statistics I*, pages 389–399. Addison-Wesley Publishing Company, 1986.

[18] Daryl Pregibon. Incorporating statistical expertise into data analysis software. In *The Future of Statistical Software*, pages 51–62. National Research Council, National Academy Press, 1991.

[19] William B. Rouse, Norman D. Geddes, and Renwick E. Curry. An architecture for intelligent interfaces: Outline of an approach to supporting operators of complex systems. *Human-Computer Interaction*, 3:87–122, 1987.

[20] John Sall and Ann Lehman. *JMP Start Statistics: A Guide to Statistics and Data Analysis Using JMP and JMP IN Software*. Duxbury Press, 1996.

[21] Robert St. Amant. *A Mixed-Initiative Planning Approach to Exploratory Data Analysis*. PhD thesis, University of Massachusetts, Amherst, 1996. Also available as technical report CMPSCI-96-33.

[22] Robert St. Amant and Paul R. Cohen. Control representation in an EDA assistant. In Douglas Fisher and Hans Lenz, editors, *Learning from Data: AI and Statistics V*, pages 353–362. Springer-Verlag, 1996.

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

[24] Robert St. Amant and Paul R. Cohen. Evaluation of a semi-autonomous assistant for exploratory data analysis. In *International Conference on Autonomous Agents*, 1997.

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

[26] Joseph W. Sullivan and Sherman W. Tyler. *Intelligent User Interfaces*. ACM Press, 1991.

[27] L. G. Terveen. Intelligent systems as cooperative systems. *International Journal of Intelligent Systems*, 3(2–4):217–250, 1993.

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

[29] Luke Tierney. *LispStat: An Object-Oriented Environment for Statistical Computingand Dynamic Graphics*. John Wiley & Sons, Inc., 1991.

[30] John Tukey. An alphabet for statisticians' expert systems. In W.A. Gale, editor, *Artificial Intelligence and Statistics I*, pages 401–409. Addison-Wesley Publishing Company, 1986.

[31] John W. Tukey. *Exploratory Data Analysis*. Addison-Wesley Publishing Company, 1977.

[32] Forrest W. Young and David J. Lubinsky. Learning from data by guiding the analyst: On the representation, use and creation of visual statistical strategies. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, pages 531–539, 1995.