A planning perspective on strategic data analysis

Robert St. Amant

Department of Computer Science
North Carolina State University
EGRC-CSC Box 7534
Raleigh, NC 27695-7534
stamant@csc.ncsu.edu
http://www.csc.ncsu.edu/eos/users/s/stamant/www/index.html

Abstract. Over the past fifteen years a variety of interactive IDA systems have been developed to support strategic reasoning for data analysis. We present an AI framework, hierarchical task network planning, that provides a unifying view of these efforts. We describe a taxonomy of strategic approaches to IDA, based on the planning framework, and discuss the strengths and weaknesses associated with each approach.

1 Introduction

A number of researchers have constructed explicit models of the data analysis process, usually with an eye toward automating some portion of it. Oldford and Peters describe data analysis in terms of operational levels [18]. At the highest level in a hierarchy of abstraction, we have strategies for dealing with general concerns, such as designing experiments or deciding how to analyze archival data. The decisions and actions we need to consider at each level expand into a set of new tasks that depend on capabilities at the next lower level. As we descend the hierarchy, these decisions rely less on abstract knowledge and more on rules or guidelines that can be formalized and automated. Just above the lowest level we find statistical procedures, such as regression and its associated analyses; the very lowest level contains the most primitive elements of a statistical computing environment, routines for numerical processing and display. Others descriptions of the levels of data analysis have been proposed, all largely compatible [3, 21, 25].

Hand gives a different account, describing data analysis as four stages [11]. In the first stage we formulate the general aims of the analysis. This stage is critical to success: in it we decide which variables should be considered, how they are related, and which research questions are relevant. In the second stage our aims are translated into the formal terms of a set of statistical modeling techniques. The third stage is numerical processing. The final stage inverts the original aims formulation effort, to determine how the statistical results translate to interpretations and actions in the real world. Like levels, stages may contain more detailed stages: data cleaning, transformation, and estimation are all components of the numerical processing stage. Others have proposed similar descriptions of data analysis as stages [1, 2], differing mainly in the level of granularity.

These two perspectives can be combined into a single account of data analysis [9]. Stages at a high operational level are abstract tasks that we must translate into procedures we can actually carry out. The stage-based description imposes a temporal or sequential ordering on these abstract tasks. To execute a task, we turn to knowledge about operational levels, which tells us how to break a task down into more detailed components at a lower level of abstraction. These components are processed in the same way, their ordering governed by stages, their decomposition by knowledge about levels.

Our goal in this paper is to show how a specific AI formalism, hierarchical task network planning, captures this integrated view of the structure of data analysis. We concentrate on implemented systems that assist users with strategic, decision-making aspects of the process. We describe various approaches to the problem of strategic data analysis and discuss their strengths and limitations. Our own current research involves VIA, a data analysis and visualization assistant based on the framework presented in this paper, but a discussion of this work, which is at a preliminary stage, is beyond the scope of this paper.

2 Data analysis and planning

Hierarchical task network planning is based on the notion of task networks, sets of tasks that represent things to be done. In our discussion we will use the terms "plan" and "task network" interchangeably. *Operators* are tasks that can be executed directly: computing a mean or generating a scatter plot would be reasonable operators for a statistical HTN planner. *Compound operators* are tasks that specify a network of other tasks, along with constraints on how the tasks are to be executed. A compound operator for a statistical planner could specify that a numerical processing task should expand to an ordered sequence of data cleaning, transformation, and estimation tasks, for example. Finally *goals* are special tasks that describe some desired state of the world without specifying how it is to be reached. A statistical planner might establish the goal of having a model with some particular properties, and let further task expansions determine exactly how the model is to be generated. An HTN planner solves problems by selecting tasks to satisfy goals, deciding at each point whether a directly executable operator will do, which compound operator might be appropriate, how it should be expanded. The result is an network of tasks at increasing levels of detail that expands until it can satisfy a top-level goal.

A variety of approaches to strategic data analysis are encompassed in this planning framework. Figure 1 shows an overview of the general process. Each stage corresponds to a task in a network; operational levels are represented by the expansion of tasks into lower-level tasks. Tasks expand as the hierarchy deepens until computational procedures are reached at the lowest operational level.

In what we'll call the *conventional architecture* of the data analysis process, the statistical system is responsible for providing and executing the standard data analysis operators and modeling procedures. These constitute the legal operators and possibly some small set of compound operators of a hypothetical

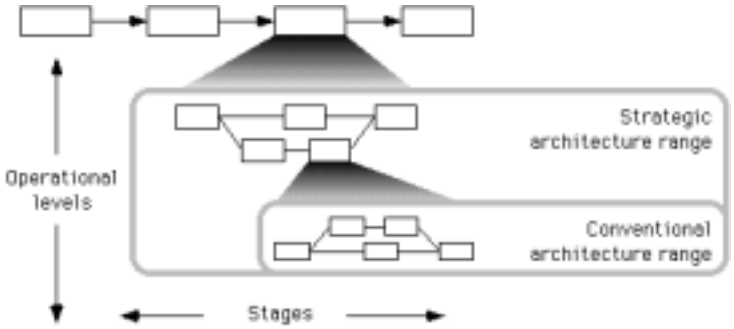

Fig. 1. A task network for data analysis

HTN planner in the data analysis domain. The system is further responsible for maintaining structures to record and organize relationships between data, derivations, descriptions, and models—the environment, or planning data, on which the tasks operate. The human analyst retains control of higher levels in the hierarchy, the more abstract, decision-making process: dynamically constructing task networks with the operators available in the system, determining how each decision is justified by properties of the data, directing the course of the analysis in general.

Strategic systems vary the conventional architecture by assuming more responsibility for the process. A system might provide a set of operators at a higher operational level than a conventional system, or generate information to motivate or justify the human analyst's decisions, or make some decisions on its own. We find that strategic systems generally cluster in the middle stages of the process and at lower operational levels. In practice, formulating aims and interpreting results can require a good deal of experience and real-world knowledge. Similarly, decisions at higher levels of abstraction often depend on statistical knowledge that has not yet been formalized, or may not be entirely understood. A strategic system can nevertheless participate in the decision-making process, as a collaborator, even if its reasoning does not lead directly to actions.

3 Approaches to strategic data analysis

We can divide strategic approaches into several categories depending on whether they address data representation, result evaluation, or procedural issues. In our discussion of each category, we describe the approach, its interpretation in the planning framework, a few representative systems, and the advantanges and disadvantages. What we hope to gain from this discussion is an understanding of how different approaches to data analysis compare with one another in a larger common context, and how they might in principle be integrated.

Data representation. Data values are conventionally stored in arrays, which supports generic formats and consistency in data manipulation. There is no overriding *statistical* reason, however, for this representation [17]; though computationally convenient, it is not ideal for capturing the real-world knowledge we might find in richer, more structured representations of data.

DINDE was an influential early system aimed at professional statisticians [17]. DINDE represents data objects in terms of `individuals`, on which measurements can be taken, `variates`, which are the measurements that can be taken, `datum` objects, which represent the actual values of variates, and `collections` and `associations`, which impose organization on the data. Each of these classes has specializations to match specific statistical concepts: variates can be continuous, discrete, or categorical; a data record is a combined collection/association that stores a set of data for either an individual or a variate, and allows collective operations on the set.

The advantages of such an approach go beyond flexibility in data representation. From a planning perspective, by raising the level of abstraction of data objects, we also raise the level at which statistical operators can be effective. The structural information in a more abstract data representation informs the application of operators far more then is possible with a generic array-based representation. The approach of improving strategy by improving data representation is reflected in modern systems such VISTA and QUAIL, among others [6, 8, 26].

Primitive operators. The set of primitive operators can strongly influence the effectiveness of a data analysis system; one of our first questions about an statistical environment is inevitably, "What functions does it include?" A strategic system can facilitate the analysis process by increasing the power or level of abstraction of the operators available, or by simply replacing an existing operator that requires human input with one that does not.

This approach is common in interactive statistical systems like those mentioned above. The AIDE system takes operator abstraction a step further [22]. In addition to conventional statistical functions, AIDE defines three higher-order operations that take functions and relationships as input: reduction, transformation, and decomposition. A reduction summarizes a relationship by mapping it to a scalar value, such as the mean of a sequence of numbers. A transformation maps a function over the tuples of a relationship; familiar transformations are the log transform and even the extraction of residuals from a linear fit. A decomposition breaks a relationship down into smaller relationships, to separate a relationship into clusters, say, or to isolate and remove outliers. These operations provide the primitives for data manipulation in an exploratory data analysis, offering surprising generality in representing common procedures.

In general, by introducing a new set of planning operators, a system reshapes the search space. In AIDE the advantages are relatively minor; some generality in operators is gained. In other systems, however, especially those that rely on interaction with the user, specializing the operators to the data analysis task can lead to better usability, easier development or extension, and even improved efficiency. Consider systems that automate some part of the data analysis process,

such as data preprocessing [4]. These refocus the decision process on operators with properties very different from those of conventional primitives.

Local search. Closely related to the introduction of novel operators to the planning process is local automated search. The search for descriptions or models is one of the most important parts of data analysis. Perhaps the majority of strategic systems are developed with this view in mind: they narrowly apply AI techniques to a specific modeling problem. In our planning interpretation, these systems replace a single task, at some specific stage and operational level, with an alternative to the operator execution or task network expansion that would ordinarily follow. Instead, the system calls an independent, external search procedure that returns a value or set of values as the result of the operator. At that point the user evaluates the results and decides how to proceed, once again in the planning framework.

Ostrouchov and Frome provide an extensive example of this approach [19]. Their system searches through a space of log-linear hierarchical models with an evaluation criterion based on deviance:

$$D_A = -2(L_A - L_0).$$

Here L_A is the log likelihood for model A maximized over its parameters, and L_0 is the same for the saturated model. Model A is considered the "best" when

$$D_A + \alpha p_A$$

is minimized, where α is either constant or the number of observations, p_A the number of parameters of A. At model-selection step in the larger data analysis process, the system is activated to generate possibilities, which it computes by searching for the set of all the best models.

This approach of substituting automated search for human search is common to many other systems. In such systems data analysis proceeds in distinct phases. The user works out an initial problem formulation, which the system then solves through a systematic search. The user reviews the results and possibly the entire procedure, selects the most promising results, and revises the problem. The system repeats its search. In our planning interpretation, the user is responsible for selecting operators for expansion or execution, while the system carries out an independent search to generate results that can contribute to that process at specific points: model selection or evaluation, in most cases. This style of interaction results in an "accommodation" of the user's judgment and subject-matter knowledge [16]. Rather than trying to derive complete and final results based on limited, machine-based representation of context, the system instead gives the user intermediate results intended to be in the neighborhood of good solutions, and lets the user do the fine-tuning to decide how to proceed.

Model evaluation. Yet another closely related approach focuses on model evaluation. The examples we give in this discussion could as easily apply to the previous section. If an IDA system performs a systematic search through some space of models, it must rely on automated evaluation criteria. These may be

comparative measures (*is Model y better than Model x?*) or absolute measures (*does Model x reach a given threshold?*)

Faraway's systems for regression strategy are representative of work in incremental model search and evaluation [5, 6]. A RAP is a regression analytic procedure that performs tests and generates repairs for such flaws in a regression model as skewness, outliers, influential points, nonconstant variance, curvature, and so forth. Each RAP is a single operator in the regression modeling task. In practice, the order in which these operators are executed is only partially constrained, and may vary according to the personal preferences of the statistician applying them. Given a dataset consisting of a regression variable and a set of predictor variables, the RAP system searches for a weighted regression based on modifications to the dataset effected by different sequences of RAPs. An initial set of RAPs is chosen. Each RAP is applied to the dataset to produce a regression model, if the result of its test is significant. The new models returned are then subject to the application of further RAPs. This process generates a directed graph in which model nodes are linked by RAP applications. Eventually, models are reached that are not changed by the application of any RAP. At each step in the search, the significance test of an individual RAP is an implicit model evaluation criterion; all the paths through the model space constitute a composite criterion.

The interpretation in the planning framework is as discussed in the previous section: the intention is to supplement human judgment with an objective evaluation criterion, applied systematically over some space of models. A large number of systems, developed in knowledge discovery in databases, scientific discovery, and machine learning, fall under the umbrella of model evaluation systems. In all such systems, an important goal is to apply a consistent, objective criterion to models or descriptions under consideration, to complement a human analyst's subjective but often better-informed judgment. In fact, the potential benefit of automated model selection is so great that it is tempting to concentrate solely on evaluation heuristics, leaving aside the broader issues of data analysis. From a strategic data analysis perspective, however, if we neglect the process as a whole, we run the risk of building systems that are actually *un*intelligent [10]—systems that encourage analysts to stop after achieving overly restricted goals.

Plan expansion. From model evaluation and local search we move to systems that give more comprehensive support for data analysis. These systems make decisions and execute operators that would ordinarily be the responsibility of the user, taking active part in the data analysis process. REX, the Regression EXpert system, is the grandfather of these systems and has led to a number of successors [7, 20, 8].

Silvers et al. have developed a statistical advisory system for biomedical researchers, which we will call BIOMED [12, 21]. The domain of BIOMED is the correct application of methods for group mean comparisons. The system guides the user in checking assumptions for comparison of means problems. It gives warnings of possible pitfalls and indicates alternative directions when problems occur. Like REX, BIOMED implements a hierarchical decision tree that guides

the user through an analysis. The BIOMED representation, however, explicitly represents levels of abstraction that closely track Oldford and Peters's operational levels. From one viewpoint, BIOMED is important for its representation of strategies for ANOVA in a practical implementation; from another viewpoint its significance is as the successful integration of a large number of ideas worked out by the developers of earlier systems for statistical strategy.

The AIDE system also integrates work from earlier systems, but draws on different sources. AIDE is explicitly an HTN planning system. AIDE's plans represent a small cross-section of heuristic techniques from the exploratory data analysis literature. These include handling outliers, a number of transformation algorithms, resistant line fitting techniques, cluster detection and analysis, and a few other elementary pattern recognition techniques. The system builds and executes plans by drawing from a library of procedures, combining them on the fly as it extracts and interprets new information about a dataset. At any point the user can force the system to make a particular decision, to backtrack, or to shift focus to another area of the analysis. AIDE resembles a navigational system in many ways, in which the space of analysis results is traversed under the shared control of the user and the system.

The central data structures in each of these systems directly represent the planning process. Building such a system thus requires the translation of statistical (and related) knowledge into an explicit representation. Different data analysis domains can be modeled in this way with varying success. For example, REX's regression strategy is considered complete for simple regression and is also applicable to multiple regression problems. BIOMED's prototyped ANOVA strategies appear to be effective and relatively comprehensive. AIDE's plans cover only a tiny subset of techniques for exploratory data analysis, but are effective despite their limitations of scope. The most significant problem faced by these systems—and it remains largely unsolved today—is the representation of strategic statistical knowledge [8, 10].

Recording. Building a plan is not the entirety of planning, just as carrying out a statistical procedure is not all of data analysis. Data analysis is interactive and constructive by its nature; in order to understand a result we often need to follow its derivation. Huber has likened data analysis to the process of writing programs [15]. One of the natural responsibilities of a strategic system is to record decisions made by the user. This is more difficult than it might sound. In the conventional architecture the system represents only the lowest level of operations, but many if not most of the decisions made by the human analyst involve operators, data, and knowledge at higher operational levels. To record decisions at these levels, the system must sometimes be able to decide where the analysis can be broken naturally into stages, which entries are in error, which justifications are necessary, and so forth.

What is needed is something closer to a laboratory assistant than to a programming environment. Huber describes the requirements of such a strategic assistant in some detail, which can be met with the help of a formal model of the steps in a data analysis session [14, 15]. In this model each step is represented

as a transformation from some set of inputs to a set of outputs. The temporal order of steps and the precedence relationships between inputs and their derived outputs lets the system infer a plausible logical ordering of steps. The system allows the user to associate properties, or hints, with each step, so that the system can automatically organize the steps in the analysis by different criteria: specific groups of hints, temporal order of hints, etc. In terms of the planning view of data analysis, the laboratory assistant infers sequential constraints and groupings of primitive operators, and makes them available to the user in the task networks of newly constructed composite operators.

Other systems take a more active approach in recording user actions and decisions by maintaining a model of the decision process, as with DINDE, REX [7, 20], and AIDE [22]. Developers of such systems walk a fine line between making unwarranted inferences and missing obvious implications. Conservative approaches, being more predictable and more easily guided, have generally met with greater acceptance. Inference in these systems thus relies most heavily on knowledge that can be derived directly from the data, with little or no assumptions or extraneous information needed. As recording techniques become more comprehensive, they begin to converge with work in navigation through complex information spaces [23]. The choices we consider at every step an analysis can be interpreted as the branching nodes in a directed graph representing decision points. Navigation is an effective metaphor for managing the generation and traversal of the graph. Navigational mechanisms in a system can potentially support the selection of specific decision points to visit, traces of decisions supporting a given result, the re-examination and evaluation of a decision, bookmarking of relevant decisions to be considered at some later time, and a variety of other possibilities.

Advice. Many researchers have observed that users, especially novices, do not necessarily want access to more sophisticated functions in a statistical system; they simply want answers. To meet this goal the system must be able to track the analysis process, at an appropriate operational level, in the relevant stages, to give advice about how to proceed. ESTES, the Expert System for Time Series analysis, provides guidance to the inexperienced time series analyst in the preliminary stages of the analysis [13]. In terms of Oldford and Peters's taxonomy, ESTES operates during the first two stages of data analysis, in which aims are formulated and refined into specific statistical terms. The system works at a relatively high operational level, however, offering only incomplete coverage for the difficult tasks. The designers of ESTES emphasize that the system does not take action on its own, but instead provides relevant information to the user, in the form of automated analyses, checks on user judgments, and so forth. ESTES interacts with the user in two distinct modes. A first, rough pass is made autonomously, without the user's input. The system presents its results, compares them with the user's expectations, and if they differ runs a set of more complex but more reliable tests. ESTES is thus unobtrusive when there are no problems with the data, but can provide additional support if problems do surface.

The IDA literature is filled with descriptions of statistical advisory systems like ESTES, most based on expert systems concepts [8] (KENS is a notable ex-

ception [].) From a planning perspective the issues here are comparable to those of recording user actions by understanding the changing context of analysis activity. Two further issues in developing a data analysis advisor are correctness and scope. Statistical knowledge can be difficult to formalize, especially where it overlaps with real-world experience and human judgment. In some successful cases, developers are able to identify a relatively small domain amenable to representation and can build an advisory system with all the benefits of an online manual, and more. The second key quickly becomes important: the system must not give plausible but irrelevant advice. Building a system that understands the scope of its own knowledge can be more difficult than building in the knowledge in the first place.

4 Conclusion

We have attempted to take a broad overview of strategic data analysis as a kind of planning, sampling from the IDA literature to illustrate the correspondences. (Additional examples would provide more support, but would also triple the length of this paper.) Our intention was not to argue that all data analysis systems should be doing planning, but rather to show that most of the work we are familar with fits naturally into the planning framework.[1] Our current work involves the testing and refinement of this viewpoint through the development of data analysis assistant, based on planning techniques, to succeed our earlier system AIDE [23, 22].

References

[1] Shamsul Chowdhury, Ove Wigertz, and Bo Sundgren. Artificial intelligence methods in data analysis and interpretation. In M. Schader and W. Gaul, editors, *Knowledge, Data and Computer Assisted Decisions*, pages 199–208, 1990.

[2] E. Dambroise and P. Massotte. Muse: An expert system in statistics. In *COMPSTAT-86*, pages 271–276, 1986.

[3] Cuthbert Daniel and Fred S. Wood. *Fitting Equations to Data*. John Wiley & Sons, Inc., 1980.

[4] A. Famili, Wei-Min Shen, Richard Weber, and Evangelos Simoudis. Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, 1(1), 1997.

[5] Julian J. Faraway. On the cost of data analysis. *Journal of Computational and Graphical Statistics*, 1(3):213–229, 1992.

[6] Julian J. Faraway. Choice of order in regression strategy. In P. Cheeseman and R. W. Oldford, editors, *Building Models from Data: Artificial Intelligence and Statistics IV*. Springer-Verlag, 1994.

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

[1] The former view, however, is plausible. Our interest in this conceptual framework first arose when we found ourselves re-implementing useful parts of existing systems (REX, TESS, and others) in the planning framework of AIDE.

[8] 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.

[9] William A. Gale and David J. Lubinsky. A comparison of representations for statistical strategies. In *Proceedings of the American Statistical Association*, 1986.

[10] David J. Hand. Intelligent data analysis: Issues and opportunities. In *Advances in Intelligent Data Analysis: Reasoning about Data*. Springer, 1997. 1–14.

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

[12] Nira Herrmann, Abraham Silvers, Katherine Godfrey, Bruce Roberts, and Daniel Cerys. A prototype statistical advisory system for biomedical researchers II: Development of a statistical strategy. *Computational Statistics and Data Analysis*, 18:357–369, 1994.

[13] P. Hietala. Inside a statistical expert system: statistical methods employed in the ESTES system. In *COMPSTAT-88*, pages 163–168, 1988.

[14] 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.

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

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

[17] R. W. Oldford and S. C. Peters. DINDE: Towards more sophisticated software environments for statistics. *SIAM Journal on Scientific Computing*, 9(1):191–211, 1988.

[18] R. Wayne Oldford and Stephen C. Peters. Implementation and study of statistical strategy. In W.A. Gale, editor, *Artificial Intelligence and Statistics*, pages 335–349. Addison-Wesley Publishing Company, 1986.

[19] G. Ostrouchov and E. Frome. A model search procedure for hierarchical models. *Computational Statistics and Data Analysis*, 15:285–296, 1993.

[20] 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.

[21] Abraham Silvers, Nira Herrmann, Katherine Godfrey, Bruce Roberts, and Daniel Cerys. A prototype statistical advisory system for biomedical researchers I: Overview. *Computational Statistics and Data Analysis*, 18, 1994.

[22] Robert St. Amant and Paul R. Cohen. Intelligent support for exploratory data analysis. *Journal of Computational and Graphical Statistics*, 7(4):545–558, 1998.

[23] Robert St. Amant and Paul R. Cohen. Interaction with a mixed-initiative system for exploratory data analysis. *Knowledge-Based Systems*, 10(5):265–273, 1998.

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

[25] John W. Tukey. Styles of data analysis, and their implications for statistical computing. In *COMPSTAT-80*, 1980.

[26] Forrest W. Young and David J. Lubinsky. Guiding data analysts with visual statistical strategies. *Journal of Computational and Graphical Statistics*, 4(4):229–250, 1995.